DESIGN METHODOLOGY FOR DEVELOPING CONCEPT INDEPENDENT ROTORCRAFT ANAYSIS AND DESIGN SOFTWARE

A Master's Thesis
Presented to
The Academic Faculty

By

Joseph Davis

In Partial Fulfillment
Of the Requirements for the Degree
Master of Science in Aerospace Engineering

Georgia Institute of Technology

December 2007

DESIGN METHODOLOGY FOR DEVELOPING CONCEPT INDEPENDENT ROTORCRAFT ANAYSIS AND DESIGN SOFTWARE

Approved by:

Dr. Daniel Schrage
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Mark Costello
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Lakshmi Sankar
School of Aerospace Engineering
Georgia Institute of Technology

Date Approved: November 12, 2007

ACKNOWLEDGEMENTS

I would like to recognize and thank the following individuals for their special assistance in the completion of this thesis:

Mr. Tim Mosig	Mrs. Karen Davis	Mr. Benjamin Davis
Dr. Daniel Schrage	Dr. Mark Costello	Dr. Lakshmi Sankar
Dr. Robert Loewy	Dr. JVR Prasad	Dr. Alan Wilhite
Mr. Jeremy Bain	Mr. Mark Moore	Mr. Michael Duffy
Mr. Wei-En Li	Mr. Doug Smith	Mr. David Whyte
Mr. Jeff Staub	Mr. Cedric Justin	Mr. Kshitij Shrotri
Mr. Chirag Talati	Mr. Nick Burgess	Ryan Letniak
Eric Beyer	CPT Eugene Jones	Mrs. Amber Stone
Dr. Sandeep Agarwal	Dr. Han Gil Chae	Mr. Emre Gündüz
Dr. Byung-Ho Ahn	Dr. Adeel Khalid	Dr. Suresh Kannan
Dr. Vitali Volovoi	Mr. Bernard Laurendeau	Mr. Sameer Hameer
Mr. Apinut Sirirojvisuth	MAJ Stephen Suhr	Mr. Sumit Mishra
Ramraj Sundararaj	Matt Harrigan	David Marcus
Teppei Morita	CPT Joe Minor	Mr. Thomas Cooper

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

a	lift curve slope
a	speed of sound in air
A	rotor disk area
A_B	rotor blade area
AHS	American Helicopter Society
AR	wing aspect ratio
B	rotor tip loss factor
c	rotor blade chord
c_1	engine partial power SFC correction exponent
CBEM	Combined Blade Element Momentum (Theory)
C_D	average airfoil drag coefficient
CFD	computational fluid dynamics
CIRADS	Concept Independent Rotorcraft Analysis and Design Software
C_L	average airfoil lift coefficient
D’	section drag
D	fuselage drag
D_{FUSE}	fuselage drag
D_{WING}	wing drag
e	wing oswald efficiency factor
E	endurance
f	equivalent flat plate drag

GUI	graphical user interface
GTPDP	Georgia Tech Preliminary Design Program
HP_{ACC}	accessory horsepower required
HP_{AV}	engine horsepower available
HP_{REQ}	engine horsepower required
HP_{TOTAL}	total rotor horsepower required
HP_{TR}	tail rotor horsepower required
ihp	induced horsepower required
IPPD	integrated product and process development
k1	engine SFC correction factor with temperature constant
k2	engine SFC correction factor with temperature constant
K_{OV}	rotor induced power overlapping constant
K_U	induced power forward flight correction factor
K_μ	profile power forward flight correction factor
L’	section lift
L_{ARM}	moment arm from main rotor to tail rotor
m'	rotor intermeshing fraction
MCP	engine max continuous power available
M_ψ	rotor blade speed in Mach
N_B	number of rotor blades
NOTAR	no tail rotor
OGE	out of ground effect
php	parasite horsepower required

Q_{MR}	main rotor torque
R	rotor radius
R	range
R/C_{MAX}	max rate of climb
R_F	ratio of fuel
R_{FA}	ratio of fuel available
RFP	request for proposal
R_{FR}	ratio of fuel required
Rhp	rotor profile horsepower required
S	wing area
SFC	specific fuel consumption
SOF	Special Operations Forces
SR	specific range
t	time
T	temperature
T	main rotor thrust
T_{TR}	tail rotor thrust
u_H	induced velocity in hover
u_i	induced velocity
u_p	parasite velocity
u_r	profile velocity
v	induced velocity
V	aircraft velocity

v_d	power off rate of descent
v_l	induced velocity of upper rotor disk of intermeshing rotors
v_u	induced velocity of lower rotor disk of intermeshing rotors
V_{PARA}	aircraft velocity parallel to rotor disk
V_T	rotor tip speed
W	aircraft weight
W_{CREW}	crew weight
W_E	empty weight
W_{EMPTY}	empty weight
W_{FA}	weight of fuel available
W_{FR}	weight of fuel required
W_{FUEL}	fuel available weight
W_G	gross weight
W_{MAX}	max gross weight
$W_{PAYLOAD}$	payload weight
x_1	engine transient power rating constant
x_2	engine transient power rating constant
z_1	engine power degradation factor with altitude and temperature
z_2	engine power degradation factor with altitude and temperature
α	angle of attack
α_{90}	angle of attack at $\psi = 90$ deg
α_{270}	angle of attack at $\psi = 270$ deg
η_{XMSN}	transmission mechanical efficiency

η_H	mechanical hover efficiency
η_F	mechanical forward flight efficiency
θ	rotor blade collective angle
θ_N	rotor nacelle angle
θ_T	rotor blade twist angle
λ'	rotor inflow
μ	rotor advanced ratio
ρ	air density
σ	rotor solidity
Φ	rotor inflow angle
Φ	empty weight fraction
ψ	rotor blade azmuthal angle
Ω	rotor angular velocity

SUMMARY

Throughout the evolution of rotorcraft design, great advancements have been made in developing performance analysis and sizing tools to assist designers during the preliminary and detailed design phases. However, very few tools exist to assist designers during the conceptual design phase. Most performance analysis tools are very discipline or concept specific, and many are far too cumbersome to use for comparing vastly different concepts in a timely manner. Consequently, many conceptual decisions must be made qualitatively. A need exists to develop a single software tool which is capable of modeling any type of feasible rotorcraft concept using different levels of detail and accuracy in order to assist in the decision making throughout the conceptual and preliminary design phases. This software should have a very intuitive and configurable user interface which allows users of different backgrounds and experience levels to use it, while providing a broad capability of modeling traditional, innovative, and highly complex design concepts.

As an illustration, a newly developed Concept Independent Rotorcraft Analysis and Design Software (CIRADS) will be presented to prove the applicability of such software tools. CIRADS is an object oriented application with a Graphical User Interface (GUI) for specifying mission requirements, aircraft configurations, weight component breakdowns, engine performance, and airfoil characteristics. Input files from the GUI are assembled to form analysis and design project files which are processed using algorithms developed in MATLAB but compiled as a stand alone executable and embedded in the GUI. The performance calculations are based primarily upon a modified momentum theory with empirical correction factors and simplified blade stall models. The ratio of fuel (R_F) sizing methodology is used to size the

aircraft based on the mission requirements specified by the user. The results of the analysis/design simulations are then displayed in tables and text fields in the GUI. The intent for CIRADS is to become a primary conceptual sizing and performance estimation tool for the Georgia Institute of Technology rotorcraft design teams for use in the annual American Helicopter Society Rotorcraft Design Competition.

CHAPTER 1: INTRODUCTION

The development of modern rotorcraft requires a highly iterative design methodology which balances and synchronizes multiple design disciplines in order to provide an optimum solution which best meets the needs and requirements of the customer. While different design methodologies may vary in detail, most begin with a conceptual design phase which is focused on understanding the problem, weighting the customer requirements, generating feasible alternatives, and comparing alternatives in order to make major conceptual design decisions. Once a concept is selected, the preliminary design phase begins. During this phase, the designers continually refine their focus to conduct more detailed analysis in order to refine design parameters across multiple disciplines. Figure 1.1 shows the Georgia Institute of Technology Rotorcraft Integrated Product and Process (IPPD) Preliminary Design Process. The upper left corner of this figure shows the conceptual design iteration loop, while the remainder of the figure is dedicated to preliminary design.

Throughout the evolution of rotorcraft design, great advancements have been made in developing analysis tools to assist designers during the preliminary and detailed design phases. Many of these tools are mentioned in Figure 1.1. However, very few tools exist to assist designers during the conceptual design phase. Most analysis tools are very discipline or concept specific, and many are far too cumbersome to use to compare vastly different design concepts in a timely manner. Consequently, many conceptual decisions must be made qualitatively. Figure 1.1 shows that information from the preliminary design loops are fed back into the conceptual loop making the process iterative. In this way, conceptual decisions may be ratified or rejected from information gathered during preliminary design. However, a poor conceptual design

decision that is not discovered until the completion of a preliminary design iteration will become very expensive to change and may jeopardize the critical path timeline of the project. This accentuates the need to make accurate conceptual design designs.

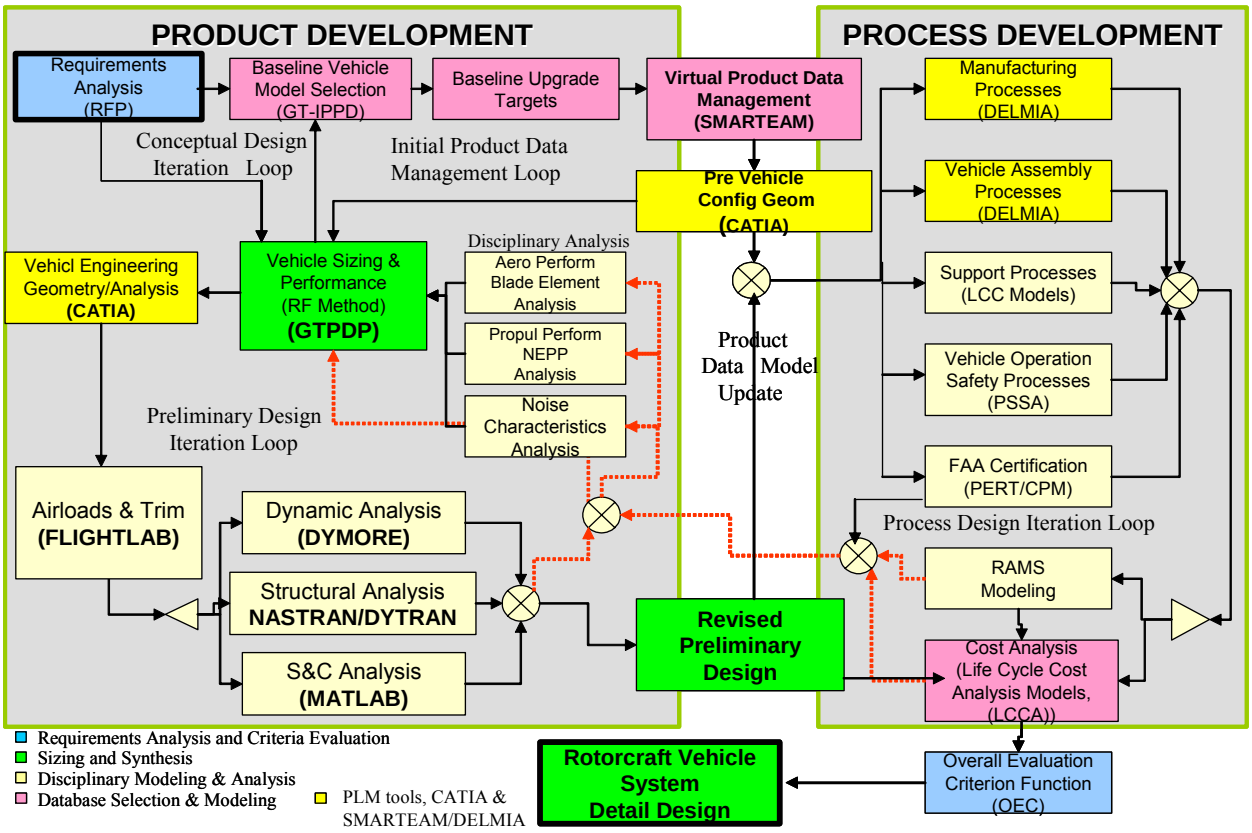


Figure 1.1: Georgia Institute of Technology IPPD Preliminary Design Process

In the next chapter, this thesis will establish the need for developing software tools which are capable of modeling any type of feasible rotorcraft design concept with different levels of detail and accuracy in order to assist in the decision making throughout the conceptual and preliminary design phases. Next, the mathematical background for analyzing and designing rotorcraft based on a modified Momentum Theory and Extended Ratio of Fuel (R_F) methodology will be presented. Then, design considerations, practices, and techniques for developing

conceptual performance analysis and design software tools will be explored. Finally, a general overview of the Concept Independent Rotorcraft Analysis and Design Software (CIRADS) that was developed in support of this thesis will be presented. In conjunction with this thesis, an operator's manual documenting the source code and describing how to use the software will be developed. Therefore, this thesis is not intended for those purposes.

CHAPTER 2: ESTABLISHING THE NEED FOR CIRADS

Many aircraft design problems begin with a specified concept configuration from the customer. For example, a customer may submit a request for proposal (RFP) to design a single main rotor helicopter with a range of 300 nm and a payload of 1000 lbs with an emphasis on minimizing life cycle cost. In this case, most of the conceptual design decisions have already been made. The customer specifically requested a single main rotor helicopter. It would not be prudent or cost effective to consider other types of configurations (i.e. tilt rotors, coaxial rotors, compounds, etc.). Engineers could begin with a ratio of fuel (R_F) sizing and optimization and proceed with the design process depicted in Figure 1.1.

However, some design problems are much more complex and require more thorough conceptual analysis in order to select a concept configuration. The 2007 American Helicopter Society (AHS) Student Design Competition is an example. The teams were required to design two different rotorcraft concepts (manned and unmanned) that are launched from a submerged submarine (50' below the water surface). These aircraft would be stored in a retrofitted space that is currently used to store 20 Triton Missiles (total volume of 44' tall x 17' wide x 95' long). The manned aircraft would be required to transport exactly two (no more, no less) special operations forces (SOF) soldiers from the submarine to a tactical objective 140 nm away, and return autonomously to the submarine for future missions.

The RFP implies that the manned aircraft must land to unimproved surfaces at the objective in order to deploy the two SOF soldiers. Multiple manned aircraft could be stored on the submarine, and each aircraft could make multiple trips to the objective. The primary mission metric was the deployment of as many SOF soldiers as possible to the objective in a six hour

window. This drove two major engineering considerations: 1) Make the aircraft as compact as possible to fit as many on the submarine as possible, and 2) make the aircraft as fast as possible, so that each could make multiple trips to the objective.

The unmanned aircraft was required to escort the manned aircraft to the objective, and remain at the objective in loiter for at least three hours in support of the soldiers at the objective. The unmanned aircraft was not required to land to unimproved surfaces (loiter only). Since both aircraft were acting in support of sensitive covert and clandestine operations, the most important design objective was to remain undetected throughout the mission. This engineering consideration of stealth was in direct opposition to the high performance and soldier transport rate of the primary mission metric. Both aircraft were required to hover OGE, which dictated some type of rotorcraft, but the RFP did not specify a configuration for either the manned or the unmanned aircraft. The design of both aircraft, the system used for launch and recovery, and the submarine retrofit were all responsibilities of the student design team.

The complex design requirements made several concept configurations seem worthy of consideration. The need for a high dash speed made a tilt rotor seem attractive for the manned aircraft, but the assumed high disk loading associated with a tilt rotor made it unattractive in terms of stealth (especially during the approach and landing to the objective, which would be the most important time). A simple single main rotor helicopter would be just the opposite (best for noise, but slow). The idea of a coaxial helicopter seemed attractive, because it would not require a tail rotor, so it could conceivably be shorter and maybe easier to store on the submarine, and if equipped with auxiliary propulsion, could be faster than a single main rotor helicopter. However, the blade vortex interaction would make it very unattractive during the descent and landing to the objective in terms of stealth. Since the unmanned aircraft was not required to land at the

objective, different considerations were necessary. Due to noise directivity, a rotor acting as a propeller (i.e. tilt rotor or tail sitter) would be perceived as quieter than the rotor of a single main rotor helicopter because the rotor acting as a propeller would direct noise outward whereas the rotor of the helicopter would direct noise down towards the objective. Also, a tilt rotor or tail sitter would be faster which would provide some added advantage even for the unmanned aircraft (quick reaction time, early reconnaissance, etc). The tail sitter was not considered for the manned vehicle due to it difficulty in landing to unimproved surfaces, and its poor crewmember ergonomics.

So, several concept configurations seemed like possibilities, and the idea that the two aircraft (manned and unmanned) would have different configurations also seemed like a possibility. Several debates among students at Georgia Tech took place about which concept was best for each aircraft. Though it eventually proved to be untrue, some students even conjectured that a tilt rotor could possibly have as low of a disk loading as a helicopter if the wings and rotors were larger. This could help reduce the noise and possibly give it a design edge over the helicopter. It would, however, reduce the packing density of aircraft on the submarine. This resulted in more debates about which engineering consideration was more sensitive to soldier deployment rate – the packing density on the submarine or aircraft speed. The launch system issues confused the situation even more. The biggest debate was still the trade between airspeed speed (deployment rate) and noise (stealth). The uncertainty about the sensitivities of these two engineering considerations drove the desire to produce quantitative calculations of performance, gross weight, and perceived noise levels of the various configurations.

However, it was quickly realized that there is no known single software tool that is capable of modeling each of the different configurations. All known software tools are used to

analyze very specific configurations. They cannot model other configurations. In fact, it was not very easy to find software that could "easily" model any configuration. Georgia Tech's baseline software known as Georgia Tech Preliminary Design Program (GTPDP), has been the program of choice for student design teams for many years. It is an older code that is based on momentum theory, but it gives values that are reasonable, especially for conceptual design. However, it can only model single main rotor helicopter and coaxial helicopters. It does not model a tilt rotor or tail sitter. Also, it does not have a very good user interface. It uses text input and output files which must be properly formatted, and the documentation for the software has not been well kept. So, each year, students undergo a trial and error learning process to figure it out. VASCOMP is a software tool used to calculate performance parameters of tilt rotors. Students made attempts to use it, but like GTPDP, its user interface is confusing and the program is quite cumbersome for new users. During most years, when a single main rotor helicopter was the obvious configuration choice for the competition, GTPDP was a painful but nonetheless useful tool, but this was not the case during the 2007 competition. Even if the team decided to go through the trouble of finding different software tools that could model individual concepts, it would require great patients in learning the software, and the results would likely be based on different calculation methods with different assumptions, so their would be some expected variance between results just based on the software used. There is also a finite amount of time that can reasonably be allocated to conceptual design during an AHS competition (or any other design scenario). Time must be allocated for preliminary design to consider areas like rotor dynamics, structures, aeroelasticity, control, life cycle cost, etc. For the 2007 competition, the students were also required to allocate time for designing a submarine retrofit and a launch and recovery system. It became very tempting to select a single main rotor helicopter for both

concepts by default “just to make it easy”. However, the students resisted this temptation and decided instead to produce their own software which is capable of modeling each of the previously mentioned configurations. In fact, the software produced for the 2007 AHS competition is a crude predecessor of CIRADS which will be discussed in detail later in this thesis. The students were not initially aware of the challenges ahead of them in producing software of this magnitude. Several of them worked an upward of 50+ hours per week for months on the AHS competition alone due to the added self imposed responsibility of producing software to model different concepts. This, of course, took emphasis away from other design disciplines, so despite winning the competition, the final product was not as good as it could have been had a decent software tool been available for performance estimation and sizing.

Through countless software iterations, several myths were eventually dismissed such as, “a tilt rotor could ever have a disk loading or gross weight as low as that of a single main rotor helicopter.” Also, through additional simulations, it was proven that high dash speed was not nearly as important as initially suggested, because the launch and refuel logistics would prove to be limiting factors on the number of aircraft that could be deployed. Due to the emphasis of stealth, a single rotor helicopter with NOTAR was selected for the manned aircraft while a tandem rotor boxed wing tail sitter (very unique) was chosen for the unmanned aircraft. Figure 2.1 shows the final configurations for each of these concepts as well as an artistic rendition of the submarine launch and recovery system.

Figure 2.1: Georgia Tech 2007 AHS Design Solution (Cypher, Dragonfly, and Barracuda)

This AHS Student Design Competition is a microcosmic example of actual aircraft design. Many of the conceptual challenges faced by the students at Georgia Tech are also faced by industry engineers. Many conceptual decisions are made on a very qualitative level, because adequate software tools are not available for quantitative analysis, and there is not enough time available to produce them. A need exists to develop a single software tool which is capable of modeling any type of feasible rotorcraft concept using different levels of detail and accuracy in order to assist in the decision making process throughout the conceptual and preliminary design phases. This software should have a very intuitive and configurable user interface which allows users of different backgrounds and experience levels to use it while providing a broad capability of modeling traditional, innovative, and highly complex design concepts.

CHAPTER 3: ROTORCRAFT PERFORMANCE ANALYSIS

Throughout this thesis, the term "analysis" will be used to describe the process of calculating the performance of an aircraft that has already been designed. Given all of the required environmental and aircraft physical parameters such as altitude, temperature, gross weight, configuration type, number of rotors, rotor diameter, tip speed, airfoil type, solidity, equivalent flat plate drag, anti-torque parameters (if any), wing parameters (if any), auxiliary propulsion (if any), engine power available, etc., the full performance envelope of the aircraft may be determined. The performance parameters determined will be max dash airspeed, max range, max range airspeed, max endurance, max endurance airspeed, max rate of climb, vertical rate of climb, hover ceiling, absolute ceiling, service ceiling, power off rate of descent, stall limitations, etc. This is analysis. While analysis techniques may be iterative in nature (i.e. CBEM), the overall process is not as iterative or systematically complex as "design".

Design (in terms of aircraft sizing) may be thought of as systematically varying key aircraft physical parameters and rerunning analysis over and over until the results of the analysis "just can't get any better", as defined by a set of customer requirements. Before an engineer can begin design, a thorough understanding of analysis techniques is required. Estimating power required for different flight modes, is arguably the most fundamental but important part of aircraft analysis. In the next section, several methods will be explored for determining aircraft power required. This will form the mathematical basis for most of this thesis.

3.1 Estimating Power Required

Several methods exist for estimating the power required for a rotor or propeller to produce a given amount of thrust under various conditions. These methods range from simple Momentum Theory to Combined Blade Element Momentum (CBEM) Theory to complex computational fluid dynamics (CFD). The major trade to consider when choosing a calculation method is processing time vs. degree of accuracy. CFD has proven to yield extremely accurate results, but the processing time for a single iteration can take several hours. In a highly iterative conceptual design environment, it could take years to reach an optimum solution. Under current computer processing state of the art, CFD is not a viable solution for most conceptual design scenarios.

On the other hand, simple Momentum Theory or "back of the envelope" takes very little time to process but does not produce a high degree of accuracy. As much as 10 percent error may arise using this method. This can be partially compensated for with empirical correction factors. For conceptual design, where general trends may be sufficient for making decisions, Momentum Theory is usually acceptable and definitely preferable to purely qualitative estimates. Momentum Theory is based on several simplifying assumptions, so caution must be used when trying to push the envelope outside the bounds of traditional design.

The industry standard today for preliminary design is the use of CBEM codes for estimating power required. CBEM codes produce more accurate results than Momentum Theory, but they do not take as long to process as CFD. However, the design of a CBEM which can model any feasible rotorcraft concept is not a trivial event, especially when the design flexibility of rotor blade parameters is increased. Allowing high levels of non-linear twist in blades can result in ill constrained equations. Couple this with the enormous differences in inflow angles

between a traditional helicopter and a tilt rotor and the complexity of the CBEM becomes quite a challenge. Several traditional CBEM assumptions and iteration schemes break down. Also, the processing time, while not nearly as restrictive as CFD, is still a hindrance during conceptual design. The students from Georgia Tech chose to implement a CBEM code into their sizing software for the 2007 AHS Student Design Competition described in Chapter 2. The processing time for a single R_F iteration for the boxed wing tail sitter configuration was about 30 minutes just to balance the fuel, not including parameter optimization. So, while CBEM seems to be appropriate for preliminary design, it is debatable whether or not it is worth the investment for conceptual design, at least during the earlier stages of configuration selection.

Due to developmental time constraints, a CBEM was not fully integrated into CIRADS. However, all of the graphical user interface (GUI) options for the CBEM were completed. A fully developed and tested MATLAB CBEM code could be integrated into CIRADS with minimal additional work to the GUI. In the next section, equations based on a modified version of Momentum Theory will be presented in detail. These equations are used for determining power required in CIRADS.

3.1.1 Momentum Theory

The nomenclature and techniques described in this section are based primarily upon the Hiller 1100 Report and the Georgia Tech Rotorcraft Aerodynamics (AE6070) class notes provided by Dr. Lakshmi Sankar [6,7]. For a traditional single main rotor helicopter, the total power required for a given flight condition is given by

$$HP_{TOTAL} = \left(ihp + Rhp + php + HP_{TR} + HP_{ACC}\right)\frac{1}{\eta_{XMSN}} \tag{3.1}$$

where,

ihp = rotor induced power

Rhp = rotor profile power

php = aircraft parasite power

HP_{TR} = tail rotor power

HP_{ACC} = accessory power

η_{XMSN} = transmission efficiency

3.1.1.1 Rotor Induced Power

From Momentum Theory, assuming a triangular inflow distribution, with two percent loss due to nonideal wake contraction and wake swirl, the induced power of a single main rotor helicopter in steady level forward flight is given by

$$ihp = \frac{1.15 K_U T}{550 B} \sqrt{\frac{T}{2\rho A}} \tag{3.2}$$

where,

T = rotor thrust required

B = rotor tip loss factor

A = rotor disk area

ρ = air density

K_U = induced power forward flight correction factor

The equation for induced power is considerably different for a tilt rotor acting as a propeller. This will be discussed after the full discussion of the single main rotor helicopter.

The rotor thrust required can be determined from a simple aircraft free body diagram shown in Figure 3.2. The thrust is a sum of the weight and drag vectors acting on the aircraft.

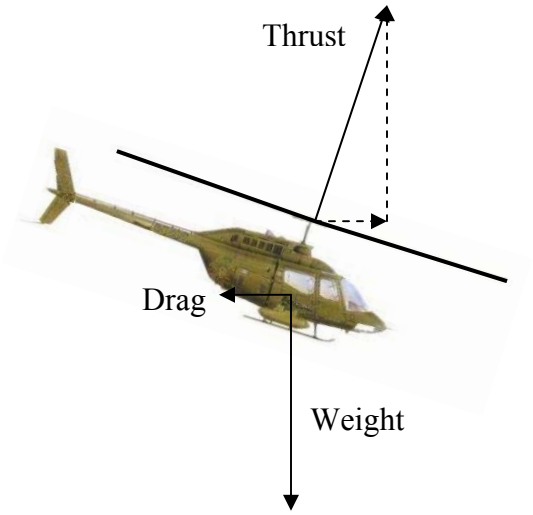


Figure 3.1: Helicopter Free Body Diagram in Forward Flight

The fuselage drag is given by

$$D = \frac{1}{2}\rho V^2 f \tag{3.3}$$

where,

V = aircraft velocity (airspeed)

f = equivalent flat plate drag

Thus, the total rotor thrust will be

$$T = \sqrt{W^2 + D^2} \tag{3.4}$$

The drag is usually considered very small relative to the weight, so some sources disregard it altogether when analyzing a single main rotor helicopter. However, for concepts that use wings and tilting nacelle angles (i.e. tilt rotors), the drag component is more significant and should be included in the thrust equation. The rotor tip loss factor is typically a value of 0.96 to 0.98 for a main rotor and 0.9 for a tail rotor. It can be approximated by

$$B = 1 - \frac{\sqrt{2C_T}}{N_B} \tag{3.5}$$

where N_B is the number of rotor blades and C_T is the thrust coefficient nondimensionalized by,

$$C_T = \frac{T}{2\rho A V_T^2} \tag{3.6}$$

where V_T is the rotor tip speed

Finally, to correct the induced power for forward flight, the induced power correction factor is given by

$$K_U = \sqrt{-\frac{1}{2}\left(\frac{V_{PARA}}{u_H}\right)^2 + \frac{1}{2}\sqrt{\left(\frac{V_{PARA}}{u_H}\right)^4 + 4}} \tag{3.7}$$

where u_H is the hover induced velocity given by

$$u_H = \sqrt{\frac{W}{2\rho A}} \tag{3.8}$$

and V_{PARA} is the free stream velocity parallel to the rotor disk resulting in an asymmetric flow between the advancing and retreating blades. For a single main rotor helicopter, this value is assumed to be the aircraft velocity (V). For a tilt rotor, it may be defined as

$$V_{PARA} = V \sin\theta_N \tag{3.9}$$

where θ_N is the nacelle angle with 0 deg assumed to be helicopter mode and 90 deg assumed to be airplane mode.

It may be noted that the value of $K_U = 1$ in hover since $V_{PARA} = 0$. It should also be noted that $K_U = 1$ for a tilt rotor with $\theta_N = 90$. In effect, the tilt rotor is acting as a single main rotor in vertical climb (no asymmetric flow). This gives insight to the purpose of the K_U term. Its purpose is to account for asymmetric flow between the advancing and retreating blades. There is no asymmetric flow between the advancing and retreating blades of a single main rotor in vertical climb or a tilt rotor in airplane mode.

3.1.1.2 Rotor Profile Power

The rotor profile power is given by

$$Rhp = \frac{\overline{C_D}\rho A_B V_T K_\mu}{4400} \tag{3.10}$$

where,

C_D = average blade drag coefficient

A_B = total rotor blade area

K_μ = profile power forward flight correction factor

The average blade drag coefficient is dependent upon the type of airfoil and is a function of average blade lift coefficient C_L, which is typically considered to be a linear function of angle of attack. The linear scalar multiple of angle of attack is known as the lift curve slope (a) and is usually approximated as 5.73 per radian. Also, C_L is approximated as

$$C_L = \frac{6C_T}{\sigma} \tag{3.11}$$

where σ is the rotor solidity

Thus, the angle of attack is calculated as

$$\alpha = \frac{C_L}{\mathrm{a}} = \frac{6C_T}{\sigma \mathrm{a}} \tag{3.12}$$

Experimental drag polar equations for a particular airfoil can then be used to calculate C_D. For example, the drag polar equation for a NACA 63-015 airfoil is

$$C_D = 0.009 + 0.3\alpha^2 \tag{3.13}$$

The rotor blade area is simply

$$A_B = N_B cR = \sigma A \tag{3.14}$$

where,

c = blade chord

R = rotor radius

A = rotor disk area

Finally, to correct the profile power for forward flight, the profile power correction factor is given by

$$K_\mu = 1 + 3\mu^2 + C_4\mu^4 \tag{3.15}$$

where μ is the advanced ratio (V/V_T)

The value of C_4 is mathematically a value of 3/8. However, wind tunnel test results have shown that C_4 should be empirically corrected to a value of 30 for a single main rotor helicopter and 5 for a compound helicopter. The value of 30 is based on an average blade C_L between 0.3 and 0.6. The value of 5 is based upon an autorotative profile (no lift being produced). For the purpose of scheduling the value of C_4 for partial lift and propulsive force of the main rotor the following interpretive function was used

$$C_4 = 25\frac{T}{\sqrt{\left(D_{FUSE} + D_{WING}\right)^2 + W^2}} + 5 \tag{3.16}$$

where,

T = thrust provided by main rotor

D_{FUSE} = fuselage drag

D_{WING} = wing drag

W = aircraft weight

Notice, that like K_U, the value of K_μ is 1 for a rotor in hover and vertical climb. Again, the forward flight correction factors are used to account for asymmetric flow between the advancing and retreating blades.

3.1.1.3 Parasite Power

Parasite power is given by

$$php = DV = \frac{1}{2}\rho V^3 f \tag{3.17}$$

3.1.1.4 Tail Rotor Power

The tail rotor power required is based on the main rotor power. The tail rotor must produce the necessary thrust to offset the torque of the main rotor. The torque of the main rotor is defined as

$$Q_{MR} = \frac{550(iHP + RHP + php)}{\Omega_{MR}} \tag{3.18}$$

where Ω_{MR} is the rotor angular velocity

The thrust required from the tail rotor is that necessary to offset Q_{MR}.

$$T_{TR} = \frac{Q_{MR}}{L_{ARM}} \tag{3.19}$$

where L_{ARM} is the distance between the main and tail rotor shafts.

Like the main rotor, the tail rotor power has an induced and profile power component that are computed using the same equations. The tail rotor does not have a parasite power component since it does not produce forward propulsive force. The induced and profile power components of the tail rotor also have induced and profile power correction terms that follow the same behavior as those of the main rotor. The tail rotor may simply be though of as a main rotor tilted 90 deg. Some design papers make simplifying assumptions about the power required by the main rotor and define a hover mechanical efficiency η_H and a forward flight mechanical efficiency η_F. These efficiency factors are used to replace the tail rotor power making equation 3.1 appear as

$$HP = (iHP + RHP + php)\frac{1}{\eta_F} \tag{3.20}$$

In this case, the forward flight efficiency factor has replaced not only the tail rotor component of power, but also the accessory power, and the transmission mechanical efficiency. During design, this keeps engineers from being bogged down in the details of the tail rotor which are not really sensitive enough to be considered in detail during conceptual design. Typical values for η_H and η_F are 0.86 and 0.89 respectively.

Sometimes, vertical fins will be used on aircraft to offload the tail rotor during high speed forward flight. In this case, the vertical fin may be thought of as a wing turned on its side producing lift to offset the load of the tail rotor in the same way that a wing of a compound helicopter offsets the load of the main rotor. When the vertical fin completely offloads the tail rotor, only the profile power component of the tail rotor will remain with a C_4 value of 5.

3.1.1.5 Accessory Power

Accessory power is used to account for components such as generators, tachometers, pumps, etc. which are driven by the main rotor. A typical value for accessory power of a helicopter with digital instruments and a moderate level of electronic equipment is 10HP.

3.1.1.6 Correcting Induced Power for a Tilt Rotor

For a tilt rotor in airplane mode and steady level forward flight, the induced power term will appear as a rotor in vertical climb. From momentum theory, this is given by

$$iHP = \left(\frac{V}{2} + \sqrt{\frac{V}{2} + \frac{T}{2\rho A}} \right) T \tag{3.21}$$

where V is the airspeed of the tilt rotor and T is the thrust required to overcome the parasite drag of the fuselage and the drag of the wings.

In this case, it is assumed that the wings are producing 100 percent of the aircraft lift to counter the weight of the aircraft. It should also be noted that the parasite power of equation 3.1 has been absorbed into the induced power term of the main rotor making it

$$HP_{TOTAL} = (ihp + Rhp + HP_{ACC})\frac{1}{\eta_{XMSN}} \tag{3.22}$$

Again, this is because the thrust of the main rotor includes the parasite drag as well as the drag of the wing. The drag of the wing may be calculated by

$$D_{WING} = 0.5 C_D \rho V^2 S \tag{3.23}$$

The drag coefficient of the wings may be calculated from the C_L of wings by

$$C_D = \frac{C_L^2}{\pi A \,\mathrm{Re}} \tag{3.24}$$

where,

AR = wing aspect ration

e = wing Oswald efficiency factor

C_L may be calculated by

$$C_L = \frac{W}{0.5 \rho V^2 S} \tag{3.25}$$

However, using the equations above, there are no constraints in place to model wing stall at low airspeeds. Typically wings have a max C_L associated with the airfoil type. One method for modeling wing stall at low airspeeds is to modify equation 3.23 as follows:

$$C_D = \frac{C_L^2}{\pi A \,\mathrm{Re}} + 0.00001\left(\frac{C_L}{C_{L-MAX}}\right)^{100} \tag{3.26}$$

For C_L values less than or equal to $C_{L\text{-}MAX}$, the value of the right hand term will be negligibly small and have virtually no effect on the value of C_D. At C_L values of even slightly above $C_{L\text{-}MAX}$, the value of the right hand term will be very large causing the drag to be very large and the induced power of the rotors to be very large. While this term works well for a tilt rotor configuration, it is not necessary for a compound configuration since a compound still has a main rotor to produce lift. Instead, the lift of a compound wing should be constrained to that of the max C_L, and the remainder of the aircraft weight should be assumed to be applied to the main rotor.

One final problem remains with this method of modeling the induced power of a tilt rotor which stems back to a very fundamental Momentum Theory assumption. The assumption is that the rotor disk provides no resistive force to the air flowing axially through it. This should not be confused with the profile drag which is correctly modeled. It is the induced drag component which is not modeled properly. Figure 3.2 shows the forces acting on a rotor blade segment.

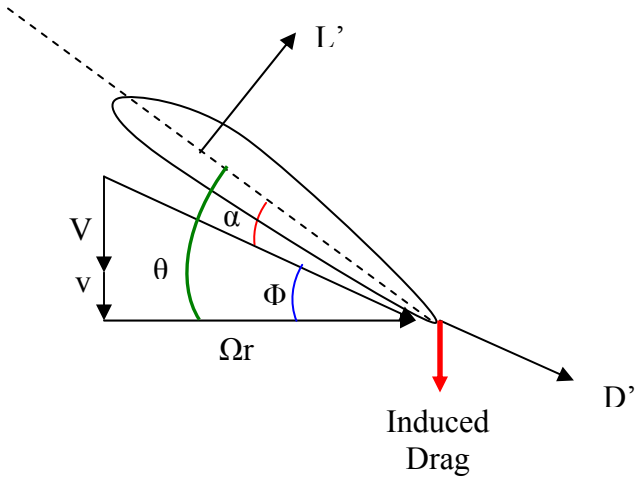


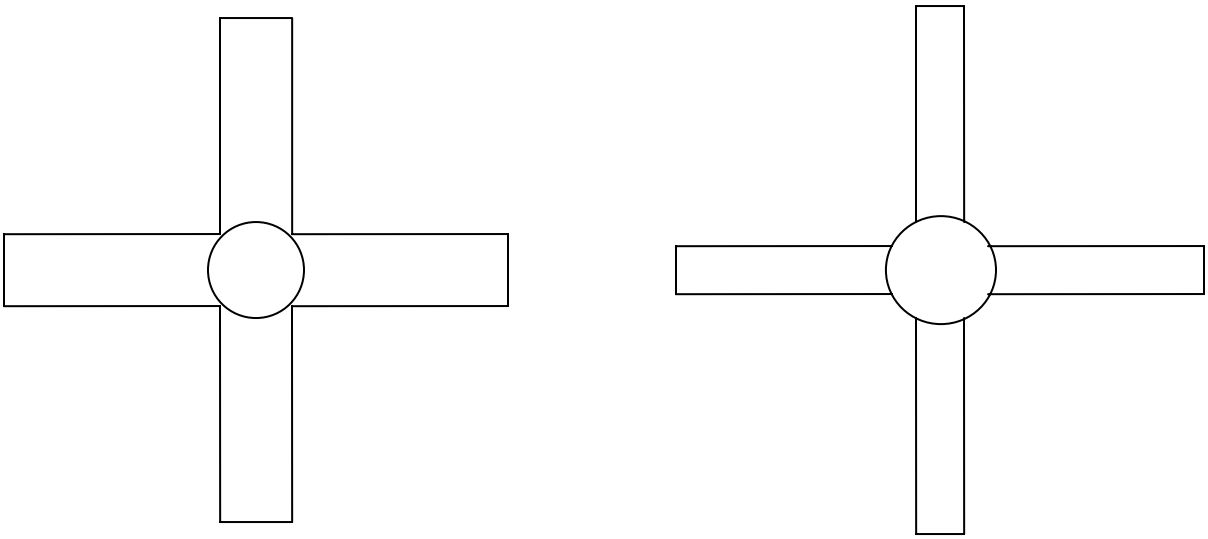
Figure 3.2: Rotor Blade Segment Forces

The induced drag component is not modeled using traditional momentum theory. For a single main rotor helicopter, this is not a problem because the induced drag force is negligible. It is negligible because the value of the axial or perpendicular flow (V+v) is very small compared to the tangential flow (Ωr). This makes the inflow angle Φ very small, which makes the induced drag very small. Even during max climb, the value of V is very small (≈ 30 ft/sec). However, during airplane mode, the value of V for a tilt rotor can be as high as 500 ft/sec, making the induced drag term significant. CBEM codes can account for induced drag accurately, but Momentum Theory must be calibrated empirically.

Upon investigation of the induced drag term using a CBEM model, it is evident that rotor tip speed has a significant effect on the induced drag term. This is the reason that tilt rotors usually have lower tip speeds in airplane mode than in helicopter mode. By slowing the tip speed, the inflow angle is increased. Also, at lower tip speeds, the blade must assume a higher angle of attack in order to produce an equal amount of thrust. Together the higher inflow angle

and higher angle of attack cause the value of collective angle (θ) to be higher (the inflow angle more so than the angle of attack). By looking at Figure 3.2 this might appear to have an adverse affect, since the total drag component will point more axial (vertical), causing the induced drag term to increase. However, because of the orientation of the rotor blade relative to the free stream velocity, the magnitude of the total drag vector is reduced. In this case, the decreased magnitude in total drag out weights the adverse change in the induced drag vector direction.

Another perspective of this phenomenon is from a position ahead of the rotor looking into the rotor as shown in Figures 3.3.a and 3.3.b. Figure 3.3.a shows a view of the blades at a higher tip speed. Figure 3.3.b shows an exaggerate view of the blades at a lower tip speed. The lower tip speed of Figure 3.3.b cause the collective angle to be higher resulting in less cross-sectional area when viewed from the front. This lower area equates to a lower total drag vector.



a) Higher tip speed b) Lower tip speed

Figure 3.3: Front View of a Tilt Rotor at Different Tip Speeds

Therefore, the empirical correction factor for induced drag should be a function of collective angle as well as the total blade area. The XV-15 was used as a calibration baseline in order to experimentally determine the induced drag correction factor. This results in the modified version of induced power.

$$ihp = \left(\frac{V}{2} + \sqrt{\frac{V}{2} + \frac{T}{2\rho A}} \right) T + \left[0.001278 \rho V^3 \sigma A (\cos\theta) \right]^{2.17} \tag{3.27}$$

Figure 3.4 shows the comparison of the XV-15 experimental data with simple and modified momentum theory. Notice that without the induced drag correction term, the error between simple Momentum Theory and the experimental values increases with airspeed.

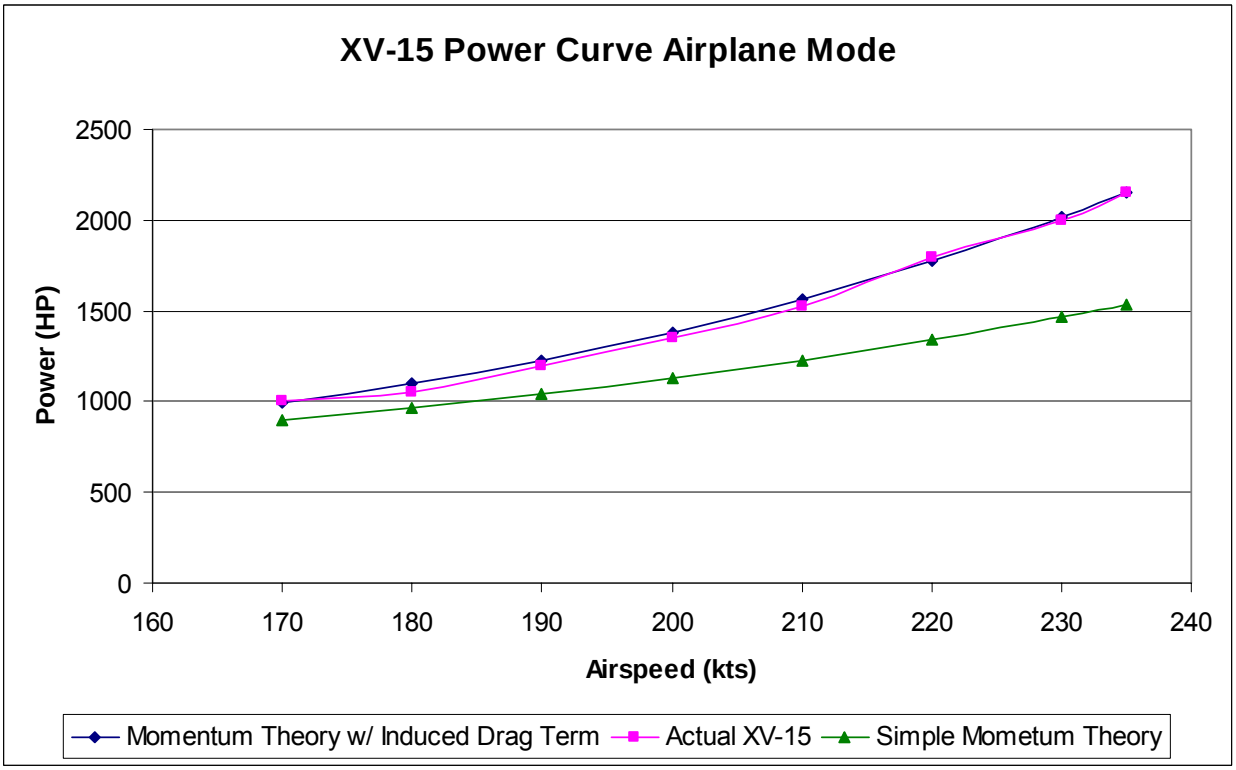


Figure 3.4: Comparison of XV-15 Power to Predicted Values using Momentum Theory with and without Induced Drag Correction Factor

3.1.1.7 Auxiliary Propulsion

The equations for an auxiliary propeller are the same as those of a tilt rotor in airplane mode. However, auxiliary propellers are usually more efficient than tilt rotors. This is because they can be optimized for high speed flight in terms of blade twist and propeller diameter. Their primary disadvantage is the empty weight that they add to the aircraft and the fact that a main rotor is still required for vertical takeoff. In fact, the main rotor is a drag hindrance during high speed forward flight due to it profile drag. So, the efficiency advantage gained from the auxiliary propeller is usually negated by the profile drag of the main rotor.

3.1.1.8 Intermeshing Rotors

The nomenclature and techniques described in this section are taken from the method presented by Prof. J. Gordon Leishman is his text book *Principles of Helicopter Aerodynamics* [4]. Two primary scenarios exist for calculating power required for coaxial or intermeshing rotors. The first assumes that the rotors have virtually no vertical gap between them. The second assumes that the lower rotor is in the far wake of the upper rotor. In either case, the ultimate goal is to define an induced power overlapping correction factor (K_{ov}). This is the ratio of the induced power required for a set of intermeshing rotors to the power required for the same set of rotors if they were not intermeshed at all. For the first scenario, assuming virtually no vertical gap between rotors, K_{ov} is defined as

$$K_{ov} = 1 + \left(\sqrt{2} - 1\right)m' \tag{3.28}$$

where m' is the overlapping fraction of the rotors, with 1 being completely overlapped (coaxial) and 0 being two separate rotors with no overlap.

Notice that in the case of a coaxial rotor (with m' = 1) that $K_{ov} = \sqrt{2}$, meaning that a set of coaxial rotors would require 1.414 times more induced power than a pair of non-overlapping rotors.

For the case of the lower rotor in the far wake of the upper rotor, the relationship between the induced velocity of the of the lower and upper rotor have the following quadratic relationship

$$\left(\frac{16m'^2}{2m'+1} - 4m' - 2\right)v_u^2 + \left(\frac{12m'}{2m'+1}\right)v_u v_l + \left(\frac{2}{2m'+1}\right)v_l^2 = 0 \tag{3.29}$$

This equation can be solved with the known value of m' to get an equation of the form

$$v_l = G(m')v_u \tag{3.30}$$

Finally, the induced power overlapping correction factor can be found by

$$K_{ov} = \frac{G(m') + 1 + 2m'}{2} \tag{3.31}$$

For the case of coaxial helicopter K_{ov} = 1.28.

3.2 Estimating Power Available

The engines in most modern rotorcraft fall into two major categories: turbo-shaft and reciprocating. The primary advantages of the turbo-shaft engine are light weight and reliability. The advantages of the reciprocating engine are low cost and lower SFC in partial power settings. The reciprocating engines are also made in smaller sizes than turbo-shaft engines giving them a "nitch" market in smaller helicopters like the Robinson R-22. This prompted the design problem

for the 2006 AHS Student Design Competition sponsored by Bell Helicopter to design an affordable turbo-shaft driven two place training helicopter. The lower SFC in partial power settings makes the reciprocating engine attractive for missions that require long loiter times such as unmanned aerial surveillance systems. However, the added weight of the reciprocating engine makes it unattractive for most other missions. In the following sections, general performance equations of a turbo-shaft engine will be explored. Due to time constraints, no reciprocating engine model has yet been added to CIRADS.

3.2.1 Turbo-shaft Engine Performance

General characteristics of a turbo-shaft engine include the following:

1. The power available based on thermal limiting decreases with higher altitude and higher ambient temperatures.
2. Power available decrease and SFC degrades (increase) with a decrease in engine RPM.
3. SFC degrades (increases) during partial power settings. The effect becomes worse as the power is lowered further.
4. SFC is relatively constant with pressure altitude, but it is sensitive to temperature. The SFC improves (decreases) at lower temperatures. Therefore, as a result of the standard lapse rate in temperature with altitude, the SFC improves with altitude. However, this is a result of the lower temperature, not higher altitude.
5. SFC values are higher (worse) for smaller engines. However, an engine's SFC is degraded more with partial power than it gains with an increase in size, so over-sizing an engine would not be advantageous.

The engine power available decreases with power available according to the following equation:

$$HP_{AV} = HP_{ISA/SLP}\left[1 - z_1\left(\frac{ALT}{10000}\right)\right]\left[1 - z_2\left(\Delta T_S\right)\right] \tag{3.32}$$

where z1 and z2 are constants on the order of 0.195 and 0.005 respectively and ΔT_S is the difference in temperature from off standard.

Transient power ratings are approximated as

$$HP_{SR} = (1 + x_1 e^{x_2 t}) \tag{3.33}$$

where HP_{SR} is the transient power rating for t minutes and $x1$ and $x2$ are constants on the order of 0.252 and -0.0173 respectively.

SFC degrades with partial power according to the following equation

$$SFC_{PART} = SFC_{MCP}\left[0.865 + \frac{0.135}{\left(HP_{REQ} / HP_{AV}\right)^{c1}}\right] \tag{3.34}$$

where c_1 is a constant on the order of 1 to 2.

SFC changes with temperature according to the following equation

$$SFC_{MCP/T2} = SFC_{MCP/T1}\left(k_1\left(T_2 - T_1\right)^2 + k2\left(T_2 - T_1\right) + 1\right) \tag{3.35}$$

where $k1$ and $k2$ are constants on the order of 0.0000218 and 0.000453 respectively

SFC degradation with decreasing RPM follows a polynomial regression that varies between engines. The scalable engine given in the 2007 AHS Student Design Competition RFP varied according to the following equation:

$$SFC_{RPM2} = SFC_{RPM1}\left[-1.211\left(\frac{RPM2}{RPM1}\right)^3 + 4.281\left(\frac{RPM2}{RPM1}\right)^2 - 5.104\left(\frac{RPM2}{RPM1}\right) + 3.034\right] \quad (3.36)$$

The horsepower of the same engine degraded with decreasing RPM according to the following equation:

$$HP_{RPM2} = HP_{RPM1}\left[1.143\left(\frac{RPM2}{RPM1}\right)^3 - 3.907\left(\frac{RPM2}{RPM1}\right)^2 + 4.58\left(\frac{RPM2}{RPM1}\right) - 0.816\right] \quad (3.37)$$

The SFC of this engine degraded with scaling according to the following equation:

$$SFC_{HP2} = SFC_{HP1}\left[\frac{-0.00932\left(\frac{HP2}{HP1}\right)^2 + 0.865\left(\frac{HP2}{HP1}\right) + 0.445}{\left(\frac{HP2}{HP1}\right) + 0.301}\right] \quad (3.38)$$

Table 3.1 shows the unscaled engine data for the 2007 AHS Student Design Competition for three different altitudes and temperatures. Notice the decrease of power available with an increase in altitude and temperature. Also notice, the improvement (decrease) in SFC with lower temperatures.

Table 3.1: Engine Rating for Unscaled 2007 AHS Design Competition Engine

Unscaled Engine Data 0ft/59F	Time Rating (min)	Power (HP)	SFC (lb/hp-hr)
OEI	0.5	1049	0.36
Max Rated Power	2	1002	0.361
Intermediate Rated Power	30	934	0.365
Max Continuous Power	MCP	764	0.379
Partial Power (50% MRP)	PRP	501	0.426
Idle	Idle	200	0.672

Unscaled Engine Data 0ft/102.92F	Time Rating (min	Power (HP)	SFC (lb/hp-hr)
OEI	0.5	867	0.373
Max Rated Power	2	820	0.377
Intermediate Rated Power	30	758	0.384
Max Continuous Power	MCP	619	0.404
Partial Power (50% MRP)	PRP	410	0.466
Idle	Idle	164	0.784

Unscaled Engine Data 6000ft/95F	Time Rating (min	Power (HP)	SFC (lb/hp-hr)
OEI	0.5	707	0.371
Max Rated Power	2	664	0.376
Intermediate Rated Power	30	611	0.383
Max Continuous Power	MCP	504	0.402
Partial Power (50% MRP)	PRP	332	0.463
Idle	Idle	133	0.777

Table 3.2: Effects of Engine Scaling on 2007 AHS Design Competition Engine

Unscaled Engine Data 0ft/59F	Time Rating (min)	Power (HP)	SFC (lb/hp-hr)
OEI	0.5	1049	0.36
Max Rated Power	2	1002	0.361
Intermediate Rated Power	30	934	0.365
Max Continuous Power	MCP	764	0.379
Partial Power (50% MRP)	PRP	501	0.426
Idle	Idle	200	0.672

Scaled Engine Data 0ft/59F	Time Rating (min)	Power (HP)	SFC (lb/hp-hr)
OEI	0.5	714	0.378
Max Rated Power	2	682	0.379
Intermediate Rated Power	30	636	0.383
Max Continuous Power	MCP	520	0.397
Partial Power (50% MRP)	PRP	341	0.447
Idle	Idle	136	0.705

Table 3.3: Effects of RPM Variation on 2007 AHS Design Competition Engine

Scaled Engine Data 0ft/59F	Time Rating (min)	Power (HP)	SFC (lb/hp-hr)
OEI	0.5	714	0.378
Max Rated Power	2	682	0.379
Intermediate Rated Power	30	636	0.383
Max Continuous Power	MCP	520	0.397
Partial Power (50% MRP)	PRP	341	0.447
Idle	Idle	136	0.705

Engine RPM Scaling	Symbol	Value
RMP Ratio to Baseline	RPM_{FR}	0.58
Power RPM Correction	SHP_{CORR}	0.75
SFC RPM Correction	SFC_{CORR}	1.28

Scaled Engine Data 0ft/59F	Time Rating (min)	Power (HP)	SFC (lb/hp-hr)
OEI	0.5	532	0.484
Max Rated Power	2	508	0.485
Intermediate Rated Power	30	474	0.491
Max Continuous Power	MCP	388	0.509
Partial Power (50% MRP)	PRP	254	0.573
Idle	Idle	101	0.903

3.2.2 Reciprocating Engine Performance

Equations for the performance of reciprocating engines will not be presented in the same detail as those for the turbo-shaft engines. However, for comparison, Table 3.4 shows data for an example reciprocating engine that was used by Georgia Tech for the unmanned aerial vehicle of the AHS 2007 Student Design Competition (reference Chapter 2). Notice that unlike the turbo shaft engine whose SFC improved continually with power, the SFC of the reciprocating engine has a bucket SFC at the 50 percent partial power rating. This makes the engine very attractive for missions requiring extended loiter. However, it should be noted that this engine has a specific weight of 1.13 lb/HP between 100 and 400 HP. So, without an extended loiter requirement, the engine will never pay for its own weight in fuel weight.

Table 3.4: Reciprocating Engine Performance

Engine Data 0ft/59F		
RPM	**Power (HP)**	**SFC (lb/hp-hr)**
4000	200	0.45
3500	178	0.435
3000	156	0.42
2500	131	0.4
2000	106	0.38
1500	80	0.4
1000	55	0.42
500	20	0.5

Engine Data 0ft/102.92F		
RPM	**Power (HP)**	**SFC (lb/hp-hr)**
4000	184	0.47
3500	164	0.455
3000	144	0.44
2500	121	0.42
2000	98	0.4
1500	74	0.42
1000	51	0.44
500	20	0.5

3.3 Aircraft Performance

In the previous sections of this chapter, the mathematical groundwork was laid for performance analysis. In this section, several key performance parameters will be defined and major performance trends will be identified. A typical plot for power required (Eq 3.1) vs. airspeed for a single main rotor helicopter at a given gross weight at sea level pressure ISA is shown below.

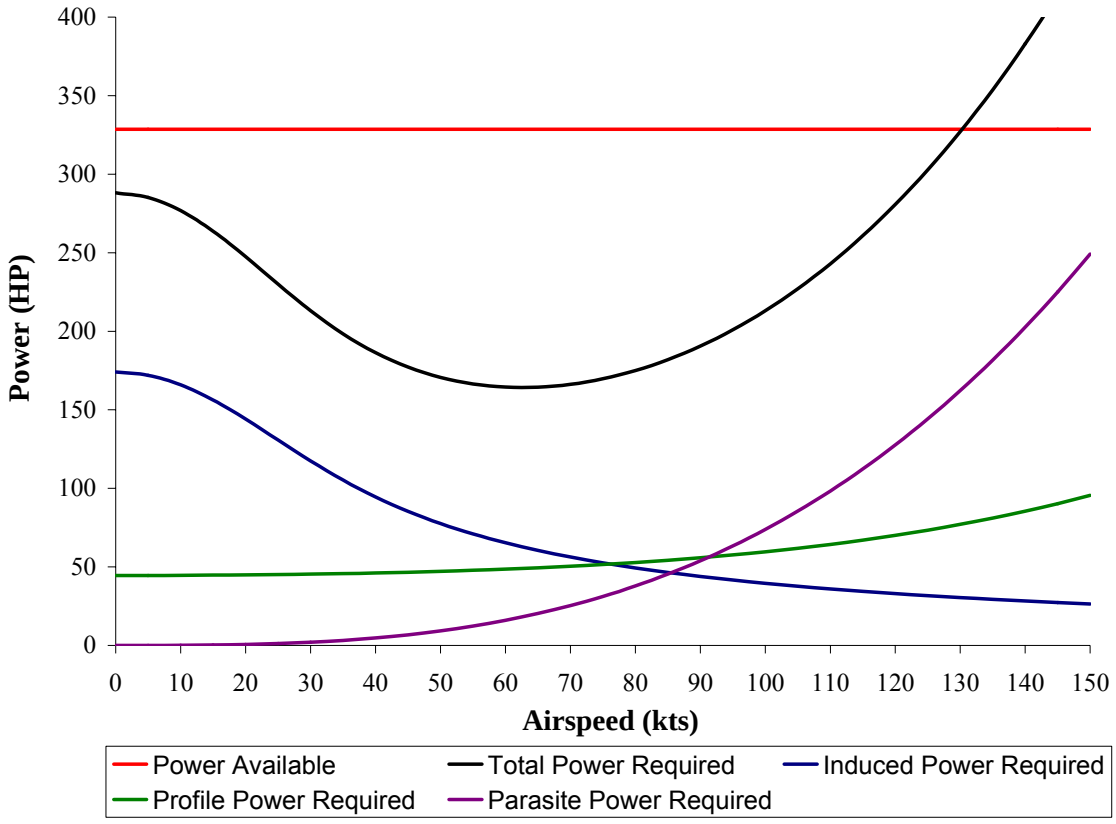


Figure 3.5: Power Required vs. Airspeed for a Single Main Rotor Helicopter

Notice that the induced power is the dominant component of power at low airspeeds and parasite power is the dominant component at high airspeeds. Also, by looking back at the equations for

induced power (Eq 3.2) and parasite power (Eq 3.3), it should be pointed out that induced power is inversely proportional to density, while parasite power is directly proportional to density. This means that it takes less power to hover at lower altitudes and it take less power for high speed forward flight at higher altitudes. The resulting plots of power required vs. airspeed for two different altitudes and temperatures are shown below on the same graph.

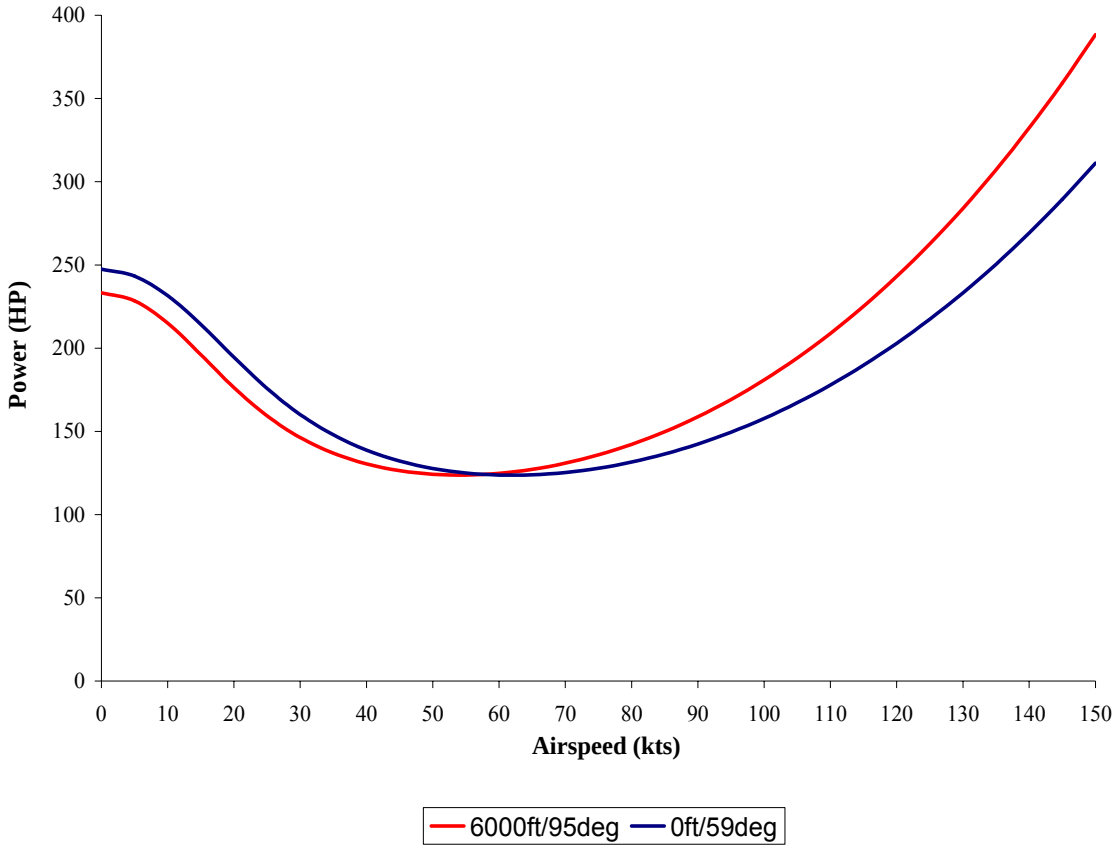


Figure 3.6: Power Required vs. Airspeed for a Single Main Rotor Helicopter at Two Different Altitudes and Temperatures

So, power required at high airspeeds decreases with an increase in altitude. However, it was mentioned in the previous section that engine power available also decreases with altitude. This

brings out an important question. Which one decreases faster? The answer is the engine. So it should be expected that the maximum airspeed for an aircraft will decrease with altitude. However, most aircraft have transmission ratings at sea level which are less than the thermal engine limits. This results because helicopters usually have high hot hover design requirements. Since it takes more power to hover high hot than at sea level, and the engine is degrade at high hot environmental conditions, the engine sizing for a high hot hover requirement results in a considerable excess power margin available at sea level. In order to reduce transmission weight, the transmission is usually sized for the high hot hover power required (not the engine power available at sea level). When the aircraft is brought back to sea level, the engine power available will increase, but the transmission power available does not, because the transmission is mechanically limited. It has nothing to do with environmental conditions. So, at sea level, the aircraft cannot take full advantage of its engine power available, because of the transmission limit. So initially, with an increase in altitude, the power available remains constant until an altitude is reached where the engine power available is equal to the transmission power rating. At higher altitudes, the engine will be the limiting power factor. Therefore, from sea level to the altitude where the engine and transmission are matched, the maximum airspeed will increase with altitude. Above this point, the maximum airspeed will decrease with altitude. A typical max altitude vs. airspeed plot is shown in Figure 3.7 below. Notice that from sea level to approximately 9000 ft, the maximum airspeed increases with altitude. Above this point, the maximum airspeed decreases. The highest point of the chart is known as the absolute ceiling. However, this plot only shows power limiting. It does not depict any stall limitations, so it does not tell the full story. Due to stall limits, the aircraft cannot fly at the higher points on this graph.

This plot will be revisited in Section 3.5 with stall limitations taken into consideration. Also note that the point where the curve crosses the y-axis is the hover ceiling.

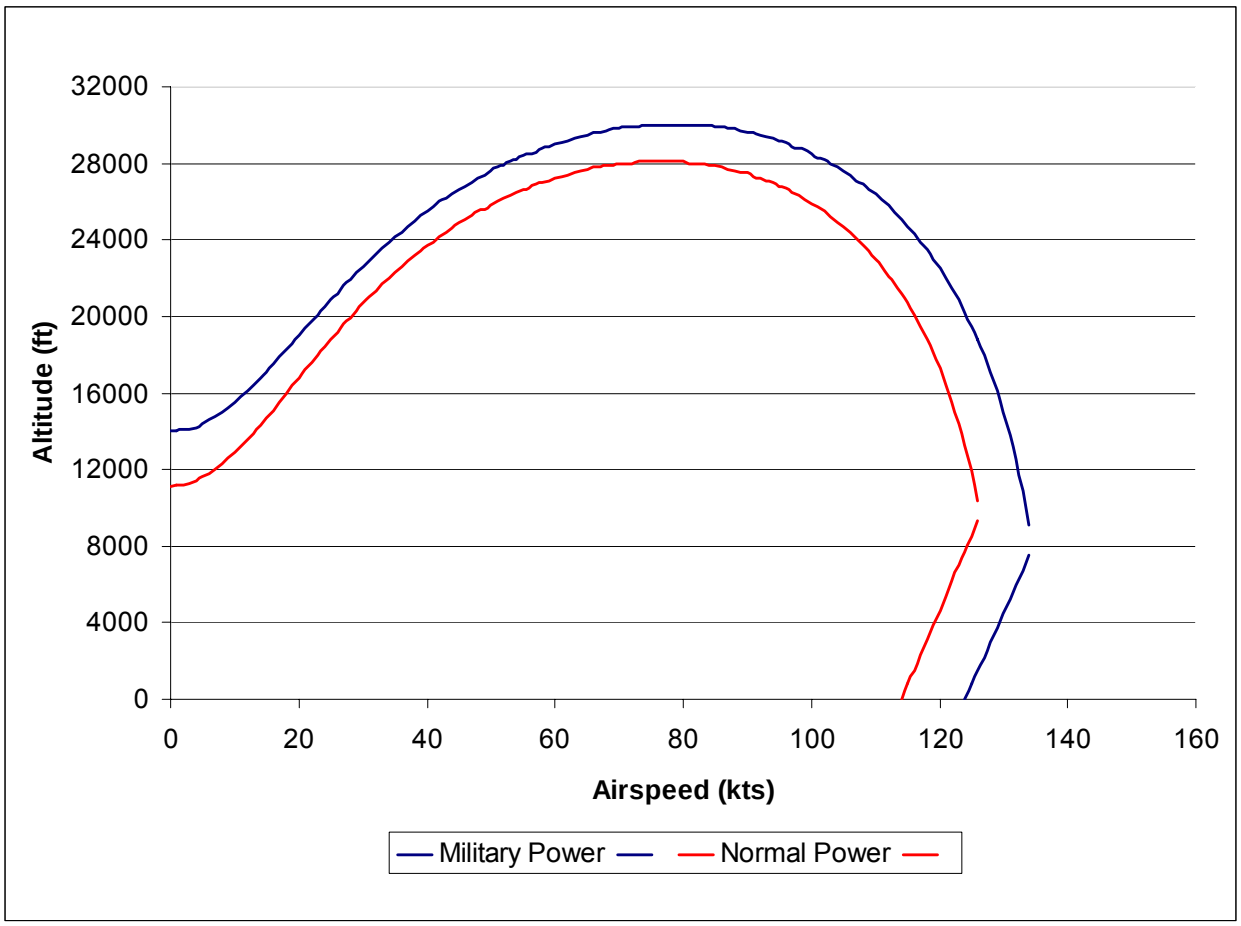


Figure 3.7: Max Altitude vs. Airspeed for Single Main Rotor Helicopter

Revisiting the power required vs. airspeed curve (shown again as Figure 3.8), three critical airspeeds should be mentioned. They are the max dash airspeed, the max range airspeed, and the max endurance airspeed. The max dash airspeed is the point on the curve where the power required curve intersects the power available line. The max endurance airspeed is the lowest point of total power on the power required curve. This is also the point of maximum rate of

climb, maximum maneuverability, and minimum power off rate of descent. The max range airspeed is shown as the point on the plot where a straight line slope leading from the origin is tangent to the power required curve. It should be pointed out that the max endurance and max range airspeeds shown on this figure are only theoretical. They assume an ideal engine perfectly designed for this particular aircraft. For real engines, as discussed earlier, the SFC values are non-linear, and in the case of turbo-shaft engines, they degrade rapidly with partial power. Therefore, the actual values for maximum range and endurance should take into consideration the SFC vs. power required for the engine.

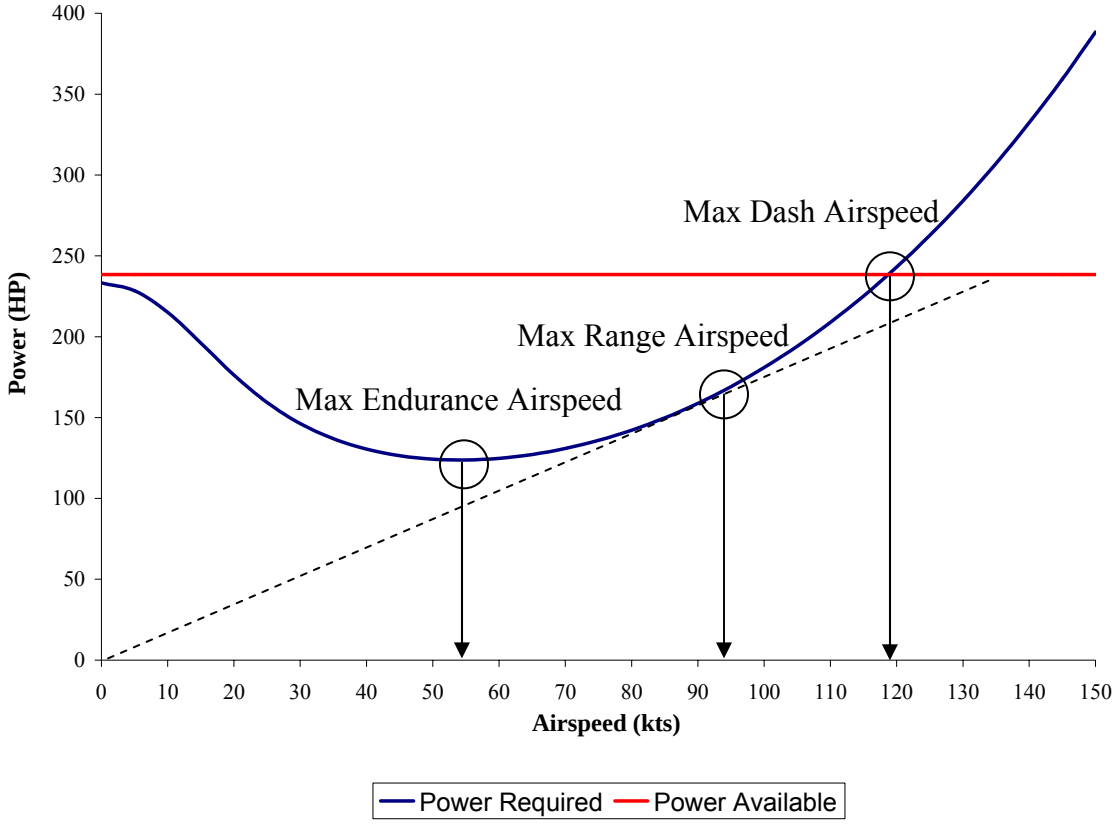


Figure 3.8 Max Dash, Max Range, and Max Endurance Airspeeds

With knowledge of engine SFC vs. horsepower (equation 3.34 may be used as an approximation), the endurance at any airspeed may be calculated by

$$E = \frac{W_{FUEL}}{SFC \times HP_{REQ}} \tag{3.39}$$

where,

W_{FUEL} = weight of fuel available on the aircraft

SFC = engine specific fuel consumption at HP_{REQ}

HP_{REQ} = power required to fly at a given airspeed

By stepping through airspeeds from a minimum to a maximum airspeed value using a reasonable airspeed step, the actual max endurance airspeed and it associate endurance are found using Eq 3.39. Similarly, the range at any airspeed may be calculated by

$$R = \frac{V \times W_{FUEL}}{SFC \times HP_{REQ}} \tag{3.40}$$

The value of range in equation 3.40 is also nondimensionalized as specific range by

$$SR = \frac{V}{SFC \times HP_{REQ}} \tag{3.41}$$

which will yield units of distance per unit weight of fuel.

Another, common airspeed associated with max range airspeed is the 99% max range airspeed. This is the airspeed just above the max range airspeed where the range is 99% that of the max range airspeed (or where the specific range is 99% of the specific range of max range airspeed). This value is generally computed because the top of the specific range curve is relatively flat.

Therefore, with a 1 percent decrease in range, the aircraft can fly up to 10 kts or so faster than the max range airspeed. Figure 3.9 shows a plot of specific range vs. airspeed with the associated max range and 99% max range airspeeds. In this case, the 99% Max Range airspeed is 12 kts faster than the max range airspeed.

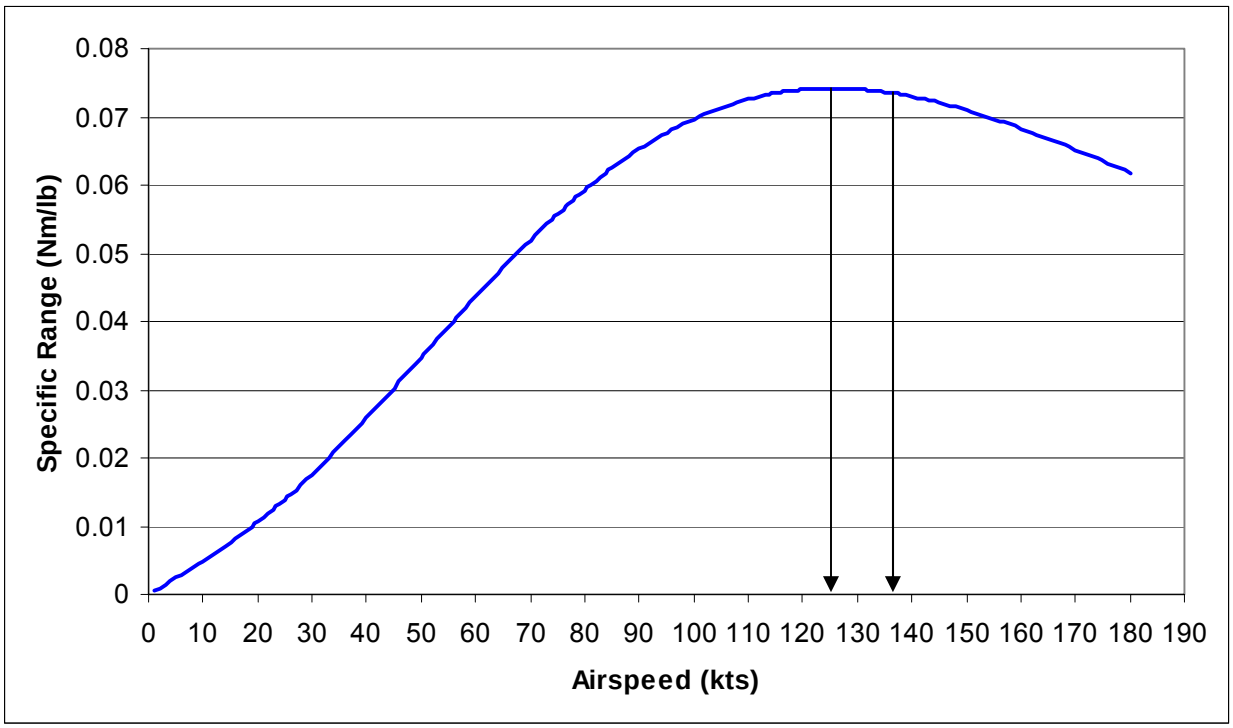


Figure 3.9: Specific Range vs. Airspeed

A point of accuracy that should be noted in using equations 3.39 – 3.41 is that while the aircraft is flying, the weight of the aircraft does not stay constant. The aircraft is burning fuel. For the purpose of analysis, there are two recommended ways of accounting for this change in weight. The simplest way is to calculate power required and SFC based on the average weight of the aircraft.

$$W = \frac{W_{MAX} + W_{EMPTY}}{2} \tag{3.42}$$

A more accurate way is to use equations 3.39 – 3.41 iteratively by recalculating the gross weight in small time increments. Starting at max gross weight, HP_{REQ} and *SFC* are computed. Using *SFC*, the fuel burned in a small time increment is calculated. This weight is subtracted from the gross weight, and HP_{REQ} and *SFC* are recalculated. The process continues until the fuel weight is zero. This gives an integrated value of fuel burn. If the fuel weight is relatively small compared to the gross weight, the accuracy gained is negligible. However, for extremely long range mission where the fuel weight may be 30 percent or more of the gross weight, the difference in accuracy may be noticeable.

It was mentioned during the discussion of Figure 3.8 that the max endurance airspeed is also the airspeed for the max rate of climb and minimum power off rate of descent airspeed. This is because the max endurance airspeed is the point of lowest total power (or very close), so it is the point with the largest power margin between power required and power available. At any airspeed, the maximum rate of climb is approximated as

$$R/C_{MAX} = \frac{33000\left(\eta_{XMSN} - \frac{HP_{TR}}{HP_{REQ}}\right)\left(HP_{AV} - HP_{REQ}\right)}{W} \tag{3.43}$$

The power off rate of descent at a given airspeed is approximated as

$$v_d = \frac{33000 \times HP_{REQ}}{W} \tag{3.44}$$

where HP_{REQ} is the power required for steady level flight at the given airspeed

3.4 Blade Stall and Compressibility

The nomenclature and techniques described in this section are taken from the Hiller 1100 Report [7]. To account for rotor blade stall and compressibility, three limitations are considered. Two of them are associated with compressibility and the other with separation stall based on a critical angle of attack. Figure 3.10 shows a single main rotor in steady level forward flight. Due to the free stream velocity (V), there is a dissymmetry of lift between the advancing and retreating blades. This results in high blade velocities at the $\psi = 90$ deg azimuth position and lower velocities at the $\psi = 270$ deg azimuth position. Also, the inboard region of the retreating blade develops a reverse flow region which produces negative lift. The lower blade velocity of the retreating blade and the negative lift of the reverse flow region cause the outboard portion of the retreating blade to operate at higher and higher angles of attack as the aircraft forward flight speed is increased. At some airspeed, the retreating blade will surpass its critical stall angle of attack. This is referred to as retreating blade tip stall. A typical critical angle of attack for a rotor blade is 12 deg.

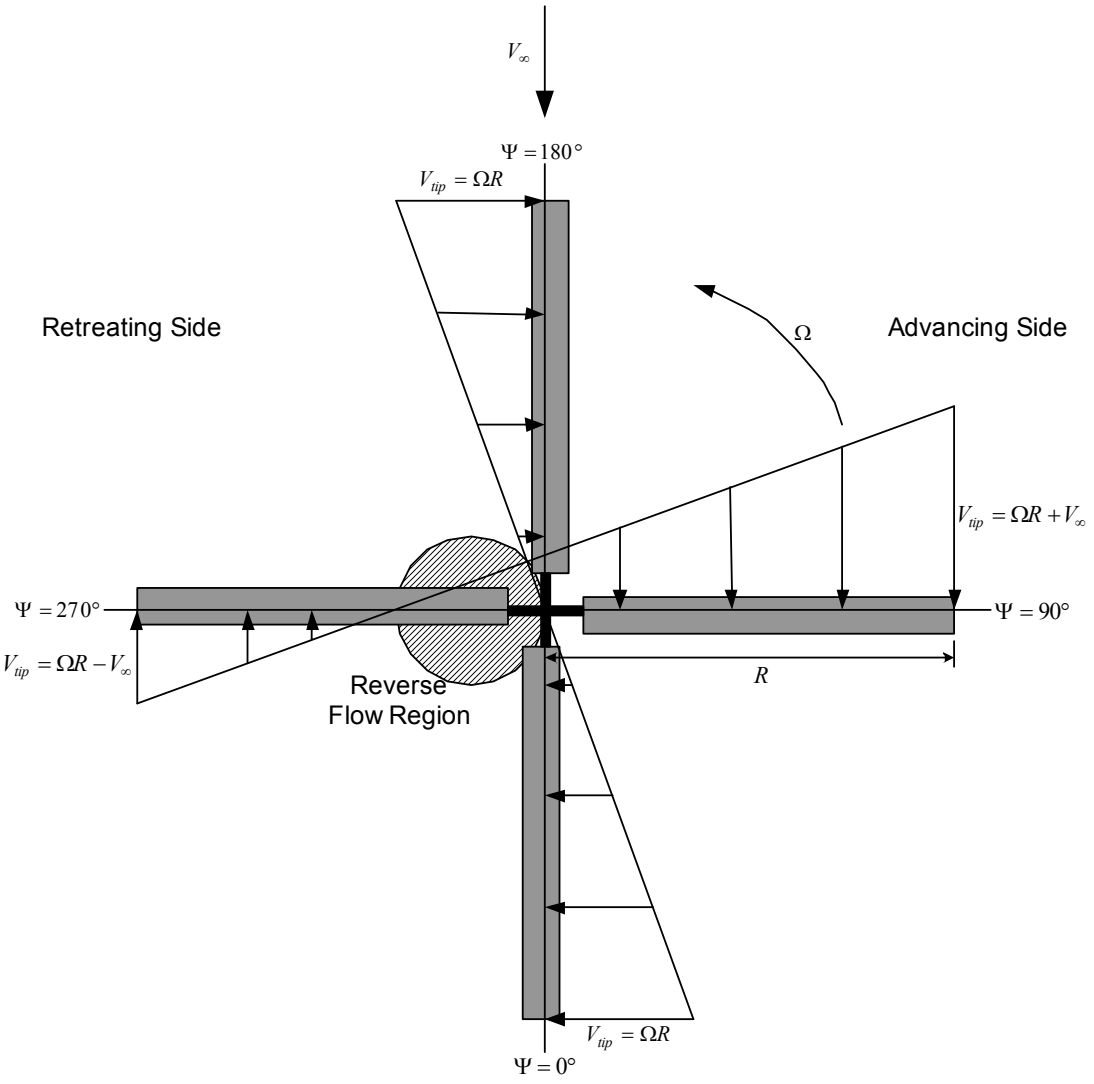


Figure 3.10 Single Main Rotor in Steady Level Forward Flight

Also, both advancing and retreating blades are susceptible to compressibility due to drag divergence. Figure 3.10 shows the drag divergence mach number vs. section angle of attack for a NACA63-015 airfoil. The retreating blade is susceptible to compressibility due to its high operating angles of attack. The advancing blade is susceptible to compressibility due to its high blade velocity.

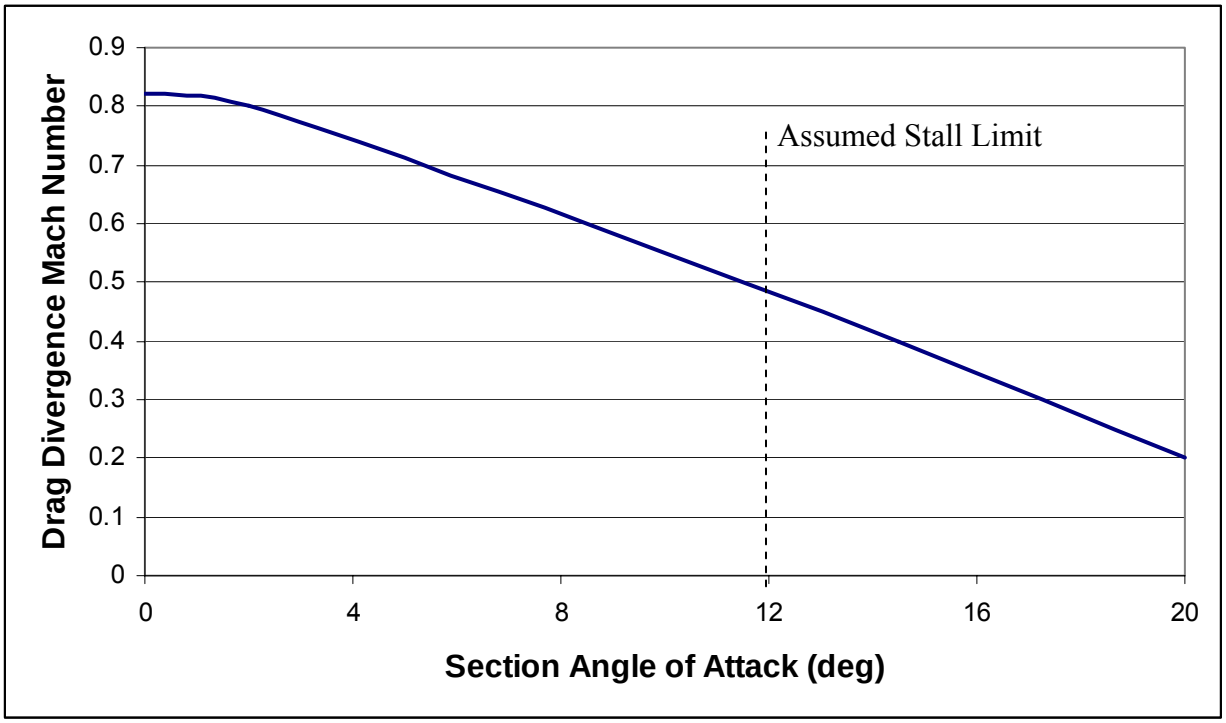


Figure 3.11: Drag Divergence Mach Number vs. Section Angle of Attack for NACA 63-015

Drag divergence will occur when the angle of attack of the blade exceeds the angle of attack of Figure 3.11 (or when the blade mach number exceeds the drag divergence mach number of Figure 3.11). The blade mach number is found by

$$M_{\psi} = \frac{V \pm V_{TIP}}{\mathrm{a}} \tag{3.45}$$

where a is the speed of sound in air

The blade angle of attacks may be found by

$$\alpha_{90} = A_1' C_L + A_2' \lambda' + A_3' \theta_T \tag{3.46}$$

$$\alpha_{270} = A_1 C_L + A_2 \lambda' + A_3 \theta_T \tag{3.47}$$

where,

$A_1, A_2, A_3, A_1', A_2', A_3'$ are constants to be defined below

C_L = blade lift coefficient from equation 3.11

λ' = rotor inflow ratio

θ_T = blade twist angle (or twist angle reflected to the tip for nonlinear twist)

Rotor inflow ratio is defined as

$$\lambda' = \frac{u_i + u_R + u_P}{V_T} \tag{3.48}$$

where,

u_i = induced velocity

u_r = profile velocity

u_p = parasite velocity

$$u_i = \frac{1}{B}\sqrt{\frac{w}{2\rho}K_u} \tag{3.49}$$

$$u_R = \frac{1100}{W}(Rhp_H)\mu^2 \tag{3.50}$$

$$u_P = \frac{550}{W}(php) \tag{3.51}$$

Constants $A_1, A_2, A_3, A_1', A_2', A_3'$ found through the following derivation

$$A_1 = \frac{1}{3a}\left(\frac{1}{K_2} + \frac{C_3}{K_2}\right) \tag{3.52}$$

$$A_2 = \left(\frac{K_1}{K_2} - C_2 + \frac{C_3 K_1}{K_2} - \frac{1}{1-\mu} \right) \tag{3.53}$$

$$A_3 = \left(1 - \frac{K_3}{K_2} - \frac{C_3 K_3}{K_2} + C_4 \right) \tag{3.54}$$

$$A_1^{'} = \frac{1}{3a} \left(\frac{1}{K_2} - \frac{C_3}{K_2} \right) \tag{3.55}$$

$$A_2^{'} = \left(\frac{K_1}{K_2} + C_2 - \frac{C_3 K_1}{K_2} - \frac{1}{1+\mu} \right) \tag{3.56}$$

$$A_3^{'} = \left(1 - \frac{K_3}{K_2} + \frac{C_3 K_3}{K_2} - C_4 \right) \tag{3.57}$$

$$K_1 = \frac{B^2}{2} + \frac{\mu^2}{4} - \frac{\left(\frac{B^2 \mu}{2} + \frac{\mu^3}{8} \right) \left(\mu B^2 - \frac{\mu^3}{3} \right)}{\left(\frac{B^4}{2} + \mu^2 B^2 - \frac{1}{6} \mu^4 \right)} \tag{3.58}$$

$$K_2 = \frac{B^3}{3} + \frac{B\mu^2}{2} - \frac{4\mu^3}{9\pi} - \frac{\left(\frac{B^2 \mu}{2} + \frac{\mu^3}{8} \right) \left(\frac{4\mu B^3}{3} + \frac{\mu^4}{6} \right)}{\left(\frac{B^4}{2} + \mu^2 B^2 - \frac{1}{6} \mu^4 \right)} \tag{3.59}$$

$$K_3 = \frac{B^4}{4} + \frac{B^2 \mu^2}{4} - \frac{\mu^4}{32} - \frac{\left(\frac{B^2 \mu}{2} + \frac{\mu^3}{8} \right) \left(\mu B^4 + \frac{1}{15} \mu^5 \right)}{\left(\frac{B^4}{2} + \mu^2 B^2 - \frac{1}{6} \mu^4 \right)} \tag{3.60}$$

$$C_2 = \frac{\mu B^2 - \frac{1}{3} \mu^3}{\left(\frac{B^4}{2} + \mu^2 B^2 - \frac{1}{6} \mu^4 \right)} \tag{3.61}$$

$$C_3 = \frac{\frac{4}{3}\mu B^3 + \frac{1}{6}\mu^4}{\left(\frac{B^4}{2} + \mu^2 B^2 - \frac{1}{6}\mu^4\right)} \tag{3.62}$$

$$C_4 = \frac{\mu B^4 - \frac{1}{15}\mu^5}{\left(\frac{B^4}{2} + \mu^2 B^2 - \frac{1}{6}\mu^4\right)} \tag{3.63}$$

With stall and compressibility calculated, the altitude vs. airspeed plot of Figure 3.7 is shows as Figure 3.12 below. As mentioned earlier, the absolute ceiling based on power alone is not truly the absolute ceiling when stall are compressibility are considered.

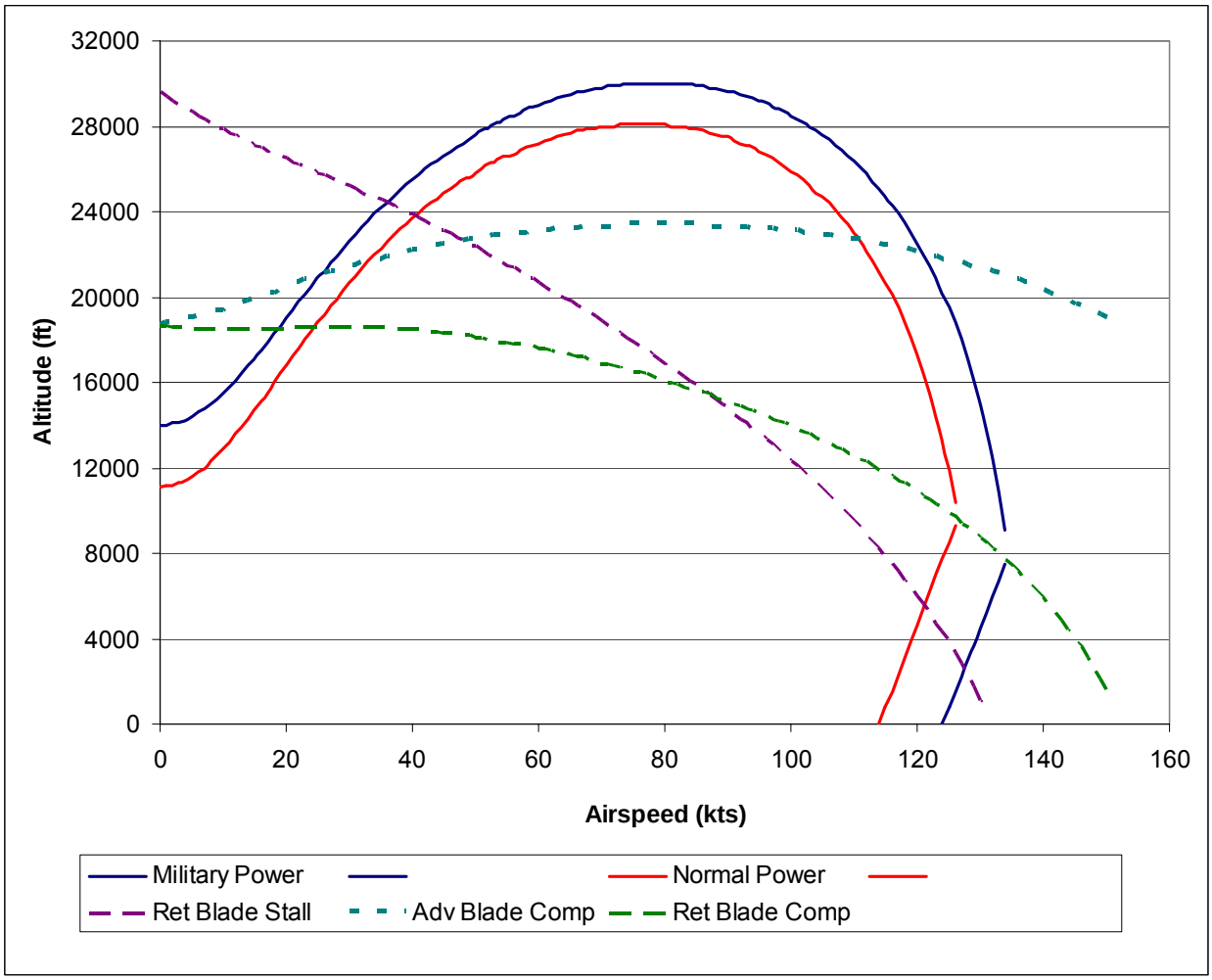


Figure 3.12 Altitude vs. Airspeed Including Blade Stall

Most of the plots in this section have been for a single main rotor helicopter, primarily because the single main rotor helicopter is the most restrictive case in terms of parameters such as blade stall and compressibility. The stall and compressibility equations also hold for any other type of configuration, but they do not always require analyzing the advancing and retreating blades separately. For example, a tilt rotor does not have a dissymmetry of lift, so the idea of blade azimuth location is meaningless. The blades will however be susceptible to both tip stall and compressibility. Figure 3.13 shows the power vs. airspeed plot of the boxed wing tail sitter

configuration designed by Georgia Tech's graduate rotorcraft design team as the unmanned aerial vehicle for the 2007 AHS Student Design Competition described in Chapter 2. The aircraft was designed to fly at different tip speeds for range and endurance. The tip speed was controlled through the engine RPM, thus the differences in power available for range and endurance. The aircraft was designed to fly at extremely low tip speeds in order to reduce the acoustic signature, so the flight envelope is quite small especially at the loiter tip speed. The spikes in power required at lower airspeeds (~75 kts) are due to wing stall. The spikes in power required at higher airspeeds are due to rotor stall.

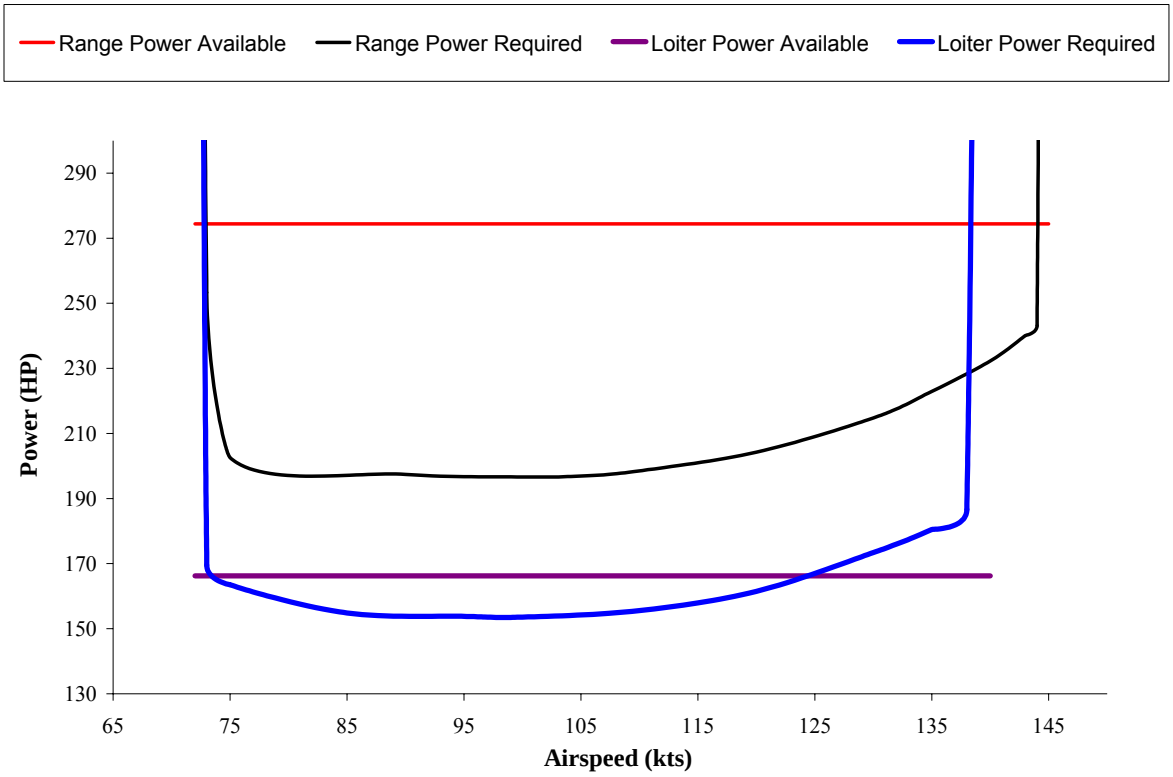


Figure 3.13: Power Required vs. Airspeed for a Boxed Wing Tail Sitter

CHAPTER 4: ROTORCRAFT DESIGN

The previous chapter established the mathematical background for analyzing rotorcraft configurations. In this chapter, techniques will be discussed for systematically and iteratively determining the best configuration for a given set of requirements.

4.1 Design Overview

In Chapter 1, the Georgia Institute of Technology IPPD Preliminary Design Process was shown as Figure 1.1. It is appropriate to show it again here as Figure 4.1 below. Assuming a design need has been established (which is the first step in any design process), the next step in the design process is analyzing the customer requirements as specified in the RFP (or implied as unspoken requirements). This will be discussed in detail in the next section. "Analyzing the customer requirements" is the blue box in the upper left hand corner of Figure 4.1. This box points to the vehicle sizing and performance using the R_F method (green box). This will be the primary focus of this chapter. The goal of vehicle sizing and performance using the R_F method is shown in the next box in Figure 4.1 as baseline vehicle model selection (pink box). This is in essence the heart of conceptual design.

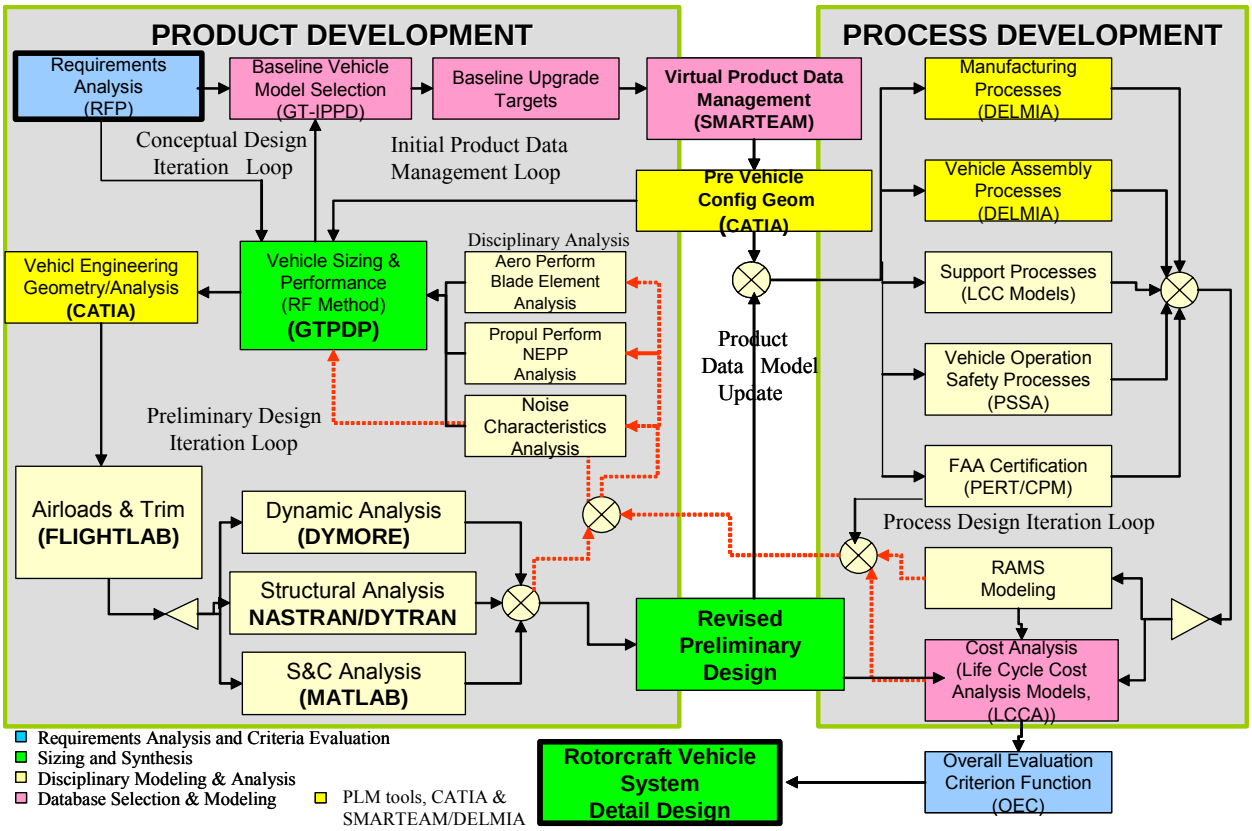


Figure 4.1: Georgia Institute of Technology IPPD Preliminary Design Process

4.2. Customer Requirements

Customer requirements are the drivers of all major design decisions. They may be classified as three types: 1) Spoken requirements, which are specifically stated in the RFP, 2) unspoken requirements, which are not specifically stated in the RFP, but they are expected by the customer, and 3) exciters, which are requirements that the customer may not have thought of, but would certainly want implemented. In terms of performance requirements, the customer is usually very specific, leaving very little room for interpretation. However, there are circumstances, even when dealing with performance requirements, that an engineer must use judgment to meet the intent of the customer when it is not specifically stated. In the case of the 2007 AHS Student Design Competition (see Chapter 2 for details), it was not specifically stated

that the manned aircraft would be required to land on unimproved surfaces, but clearly it would be necessary in order for the SOF soldiers to deploy at the object. This is an example of an unspoken requirement which affects design in terms of performance. The requirement to land on unimproved surfaces limits the disk loading to about 30 lb/ft^2 maximum. Above this, debris and rocks are prone to be washed up into the rotor system.

There are cases where customer requirements indirectly affect performance as well. In problems that require a system of systems solution, a performance parameter of the aircraft may be limited by one of the other components in the system. For example, in the 2007 AHS Student Design Competition, the requirement to store aircraft on a submarine limits the size of the rotor diameter. Carefully reading the RFP and understanding all of the customer's spoken and unspoken requirements are crucial first steps in any design solution. Afterwards, understanding the full scope of design problem in terms of all of the ancillary coordination within different components of the overall system of systems is also important. These are practices that will not be explored in detail in this thesis, but nonetheless, must occur before beginning detailed sizing and parametric design.

4.2.1 Engine Sizing Requirements

Usually, the requirements for sizing an engine will come directly from the RFP. For most rotorcraft, especially a single main rotor helicopter, a high hot hover requirement (i.e. HOGE at 6000ft/95 deg F) will be the engine sizing requirement. Sometimes, however, other requirements such as max dash speed, max rate of climb, max vertical rate of climb, max altitude, or a specific maneuverability requirement will serve as the engine sizing requirement. The one requirement which requires the most engine power will be the engine sizing requirement. The

important point to distinguish is the difference between requirements that are used for sizing the engine from the sizing mission requirements which will be discussed in the next section. The purpose of the latter is to determine the weight of fuel required (as will be discussed in Section 4.3). The sizing mission requirements are range, endurance, etc. They are not engine sizing requirements. They simply dictate how much fuel is required on the aircraft. One might argue that fuel weight determines gross weight which sizes the engine, and they would be correct. However, in the iteration scheme of the R_F method which will be discussed in Section 4.3, gross weight is assumed to be constant in each iteration (it is the step variable). Therefore, with a constant gross weight, range and endurance will not be engine sizing requirements. This will become clearer with a better understanding of the R_F method. For now, just consider engine sizing requirements to be those with pass/fail type requirements or "Max" requirements that do not pertain to fuel burn as part of the sizing mission. If no other performance requirements other than range or endurance are specified in the RFP, then by default, the engine sizing requirement will be the power required to hover OGE at the altitude and temperature of the sizing mission (or sea level pressure ISA if no environmental conditions are given). Good engineering judgment should also apply in creating a hover power margin of at least 10 percent power in this case. Missions that do not require hover at all are outside the scope of this thesis.

Engine sizing requirements may be given for a certain length of time (i.e. hover out of ground effect at 6000 ft and 95 deg F for two minutes). In this case, the short rating engine equation (Eq 3.33) will be used to equate max continuous power to the short engine rating.

4.2.2 Sizing Missions

Sizing missions are used to define customer performance requirements which are associated with range, endurance, and hover time. Sizing missions are the inputs needed in order to balance fuel using the R_F method discussed in the next section. An example sizing mission is shown in Table 4.1 below. This is the sizing mission for manned aircraft in the 2007 AHS Student Design Competition describe in Chapter 2.

Table 4.1: Sizing Mission for 2007 AHS Student Design Competition

Segment	1	2	3	4	5	6	7	Units
Type	Idle	HOGE	Cruise	HOGE	Cruise	HOGE	Reserve	-
Speed	0	0	$V_{br\text{-}99}$	0	$V_{br\text{-}99}$	0	V_{be}	ktas
Time	4	2	-	4	-	20	20	Min
Range	-	-	140	-	140	-	-	Nm
Altitude	0	0	0	0	0	0	0	ft
Temperature	102.92	102.92	102.92	102.92	102.92	102.92	102.92	°F
Engine Rating	IRP	MRP	MCP	MRP	MCP	MRP	MCP	-

4.3 Introduction to the Ratio of Fuel (R_F) Sizing Method

The purpose of the R_F method is to find the minimum gross weight configuration for a given concept. In the most primitive example, the R_F method is no more than balancing the weight of fuel required and the weight of fuel available for an aircraft that has already been designed. The weight of fuel required for a given mission is found by

$$W_{FR} = \sum \left(HP_{REQ} \times SFC \times time \right) \tag{4.1}$$

The weight of fuel for each portion of the sizing mission is calculate and summed using equation 4.1. The HP_{REQ} and SFC are found using the methods discussed in Chapter 3. The weight of fuel available is found by

$$W_{FA} = W_G - W_E - W_{CREW} - W_{PAYLOAD} \tag{4.2}$$

where,

W_G = aircraft gross weight

W_E = aircraft empty weight

W_{CREW} = crew weight

$W_{PAYLOAD}$ = payload weight

In a simple bisection algorithm of the R_F method, gross weight is used as the iterative variable. In other words the gross weight is guessed systematically until W_{FR} and W_{FA} match within some specified tolerance of say 1 lb. To set this algorithm up, a maximum upper bound for gross weight needs to be set. This will be based on engineering judgment, but when in doubt, it should be very big. Using the bisection method, the next iteration will be half that of the first if the guess is unreasonably large, so a large upper bound guess will not add much processing time. A convenient lower bound for gross weight is the zero fuel weight or operating weight.

A first guess for gross weight is the average of the upper and lower bound. This first guess gross weight is used to calculate W_{FR} and W_{FA} using equations 4.1 and 4.2. If the two are equal (or within the 1 lb tolerance), then the fuel is balanced and the guessed gross weight is the minimum gross weight. If W_{FR} is greater than W_{FA}, then the guessed gross weight is too low (more weight needs to be added in the form of fuel). In this case, the lower limit for gross weight is reset to the current gross weight guess, and the upper limit remains the same. The

second gross weight guess will be the average of the new lower limit and the preexisting upper limit. Otherwise, if W_{FR} is less than W_{FA}, then the guessed gross weight is too high (weight needs to be removed in the form of fuel). In this case, the upper limit for gross weight is reset to the current gross weight guess and the lower limit remains the same. The second gross weight guess will be the average of the preexisting lower limit and the new upper limit. This process continues until W_{FR} and W_{FA} are matched within the specified tolerance. In pseudo code this algorithm may appear as Figure 4.2 below.

```
GWmin = W_E + W_CREW + W_PAYLOAD;
GWmax = GWmin * 2;  //big number
DO
        GWguess = (GWmin + GWmax)/2;
        W_FR = GetFuelRequired (GWguess);
        W_FA = GetFuelAvailable (GWguess);
        IF (W_FA > W_FR) THEN
                GWmax = GWguess;
        ELSE
                GWmin = GWguess;
        END IF
WHILE (ABS(W_FR-W_FA) > 1);
GW = GWguess;
```

Figure 4.2: Simple Bisection Pseudo Code for R_F Method

The pseudo code in Figure 4.2 is a very simple example of the R_F method in its most primitive form. In this case, it is assumed that the empty weight remains constant and that the aircraft has already been designed. So, essentially this algorithm is just calculating how much full to put into the fuel tank of an aircraft in order to fly a certain mission. However, this is not the purpose of the R_F method. This example is just a very simple illustration to get started. The real purpose of the R_F method is to design aircraft to conduct a specific mission. This involves the systematic changing of design parameters such as disk loading, solidity, and tip speed in order to find the optimum configuration. Of these, the most sensitive is disk loading, so most R_F example use an R_{FR} vs R_{FA} with varying disk loading plot as the signature example. The terms R_{FR} and R_{FA} are simply W_{FR} and W_{FA} nondimensionalized by gross weight.

$$R_{FR} = \frac{W_{FR}}{W_G} \tag{4.3}$$

$$R_{FA} = \frac{W_{FA}}{W_G} \tag{4.4}$$

Also, it should be mentioned that the empty weight of an aircraft is highly coupled to gross weight. In the next section methods for estimating empty weight will be explored. However, for a first approximation of the R_F method, it is customary to assume an empty weight fraction (Φ) for a certain configuration. This method requires the use of "a priori" information that is representative of the configuration. Table 4.2 shows a table of "a priori" information for various concepts. Using the value of empty weight fraction, the empty weight is recalculated for each gross weight iteration in the R_F method.

Table 4.2: A Priori Design Information for Various Concepts

A Priori Design Parameters	Units	Single Main Rotor Helicopter	Coaxial Helicopter	Tilt Rotor
Disk Loading	lb/ft^2	6	10	20
Empty Weight Fraction	ND	0.55	0.6	0.65
Equivalent Flat Plate Drag	ft^2	7	9	4
Rotor Solidity	ND	0.075	0.05	0.1
Tip Speed	ft/sec	650	650	650
Downwash Factor	ND	0.03	0.05	0.08
Aux Prop Percent Thrust	ND	NA	100	NA
Wing Span	ft	NA	NA	20
Wing Aspect Ratio	ND	NA	NA	5

Figure 4.3 shows the Hiller R_F example for the design of the Hiller 1100 [7]. At the bottom of the figure is a R_F vs. gross weight plot with lines for various values of disk loading. The lines that are somewhat vertical are the curves for R_{FA} vs. gross weight. The lines that are somewhat horizontal are the curves for R_{FR} vs. gross weight. The intersections for these lines are the minimum gross weight solutions. One of these solutions, shown by the locus plot and dashed vertical line, is the absolute minimum gross weight configuration. The same method can be applied for varying other design parameters such as tip speed, solidity, wing span, wing aspect ratio, etc. However, processing time increases rapidly with each additional design parameter, so caution should be used when selecting parameters for optimization.

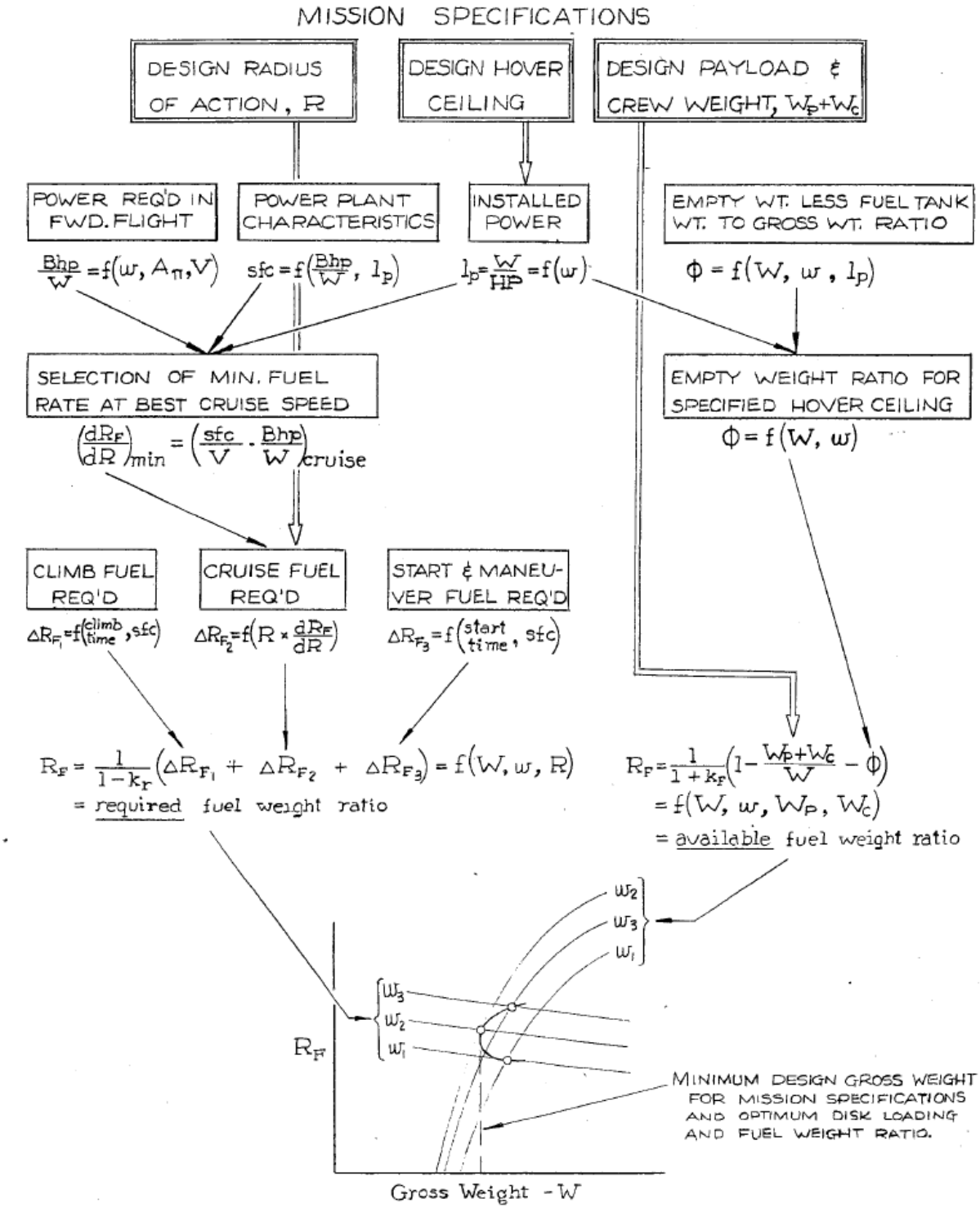


Figure 4.3: Hiller 1100 R_F Design Example

4.4 Weight Estimation

It should be obvious from the equations developed in Chapter 3 that the weight of fuel required (W_{FR}) changes with disk loading. However, it may not be obvious why the weight of fuel available (W_{FA}) changes with disk loading. W_{FA} changes with disk loading because the empty weight changes with disk loading. Estimating empty weight is one of the crudest arts performed during conceptual design. This is unfortunate, because weight is one of the most sensitive design parameters. Almost all design efforts are focused around minimizing the empty weight of the aircraft. Throughout the evolution of rotorcraft design, several empirical empty weight equations have been calibrated based on state of the art. Most empty weight equations are usually functions of physical aircraft parameters and gross weight, which means that they must be recalculated for each gross weight iteration in the R_F method. This is similar using an assumed empty weight fraction, where empty weight is also recalculated for each gross weight iteration in the R_F method. However, with empirical weight equations, the empty weight fraction will also change with each gross weight iteration resulting in different R_{FA} curves on the R_F vs. gross weight plot in Figure 4.4. For a complete listing of empty weight equations refer to the CIRADS user's manual.

CHAPTER 5: CIRADS EVOLUTION AND DESIGN CONSIDERATION

The previous chapters have established the background for rotorcraft analysis and design. In this chapter, actual implementations of the R_F method will be presented as practical examples. These implementations are predecessors of CIRADS. Advantages and disadvantages of each will be discuss, and towards the end of this chapter, the major design considerations for the development of CIRADS and other future conceptual design software tools will be considered.

5.1 RF_1 Design Example

As mentioned in Chapter 2, CIRADS was developed as a result of research conducted during the 2007 AHS Student Design Competition. CIRADS is the third iteration is a set of software tools designed with the goal of being able to model any type of feasible rotorcraft configuration. The first iteration, titled RF_1, was developed in December 2006 in an attempt to narrow possible design concepts for the 2007 AHS Student Design Competition. The software was written in Microsoft Excel with VBA macros for the iteration algorithms. It received the name RF_1 when a student saved the name of the Excel file by that name. This program was based on the first R_F method described in Chapter 4 which uses "a priori" data and assumes an empty weight fraction. The methods for calculating performance were based on the equations presented in Chapter 3, but the more refined empirical correction factors and stall models had not been determined, so there was a degree of error associated with the software. An example input section is shown as Figure 5.1 below

A Priori Design Information	Units	Symbol	Single Main Rotor Helo	Coaxial Rotor Helo	Tilt Rotor
Mech hover efficiency		η_H	0.86	0.96	0.93
Mech fwd flt efficiency		η_P	0.89	0.9	0.92
W_E / W_G		Φ	0.55	0.6	0.65
Disk Loading	lbs/ft^2	ω	6	10	20
Downwash factor	(%GW)	e_d	0.03	0.05	0.08
Figure of Merit		M	0.75	0.76	0.7
Eq Flat Plate Drag	ft^2	A_π (ft^2)	5	6	4

Simulation Inputs

Gross Weight (can be optimized with button below)	lbs	W_G	2429.6875	2917.96875	4226.5625
Rotor Solidity		σ	0.1	0.1	0.1
Rotor Tip Speed	ft/sec	V_{TIP}	650	650	650
Rotor Airfoil Selection			NACA 0015	NACA 0012	NACA 0012
Wing Span	ft	L			15
Wing Aspect Ratio		AR			5
Aux Prop percent parasite power				100	
Aux Prop Radius	ft			2.5	
Aux Prop Solidity		σ_{AUX}		0.1	
Aux Prop Tip Speed	ft/sec	$V_{TIP\ AUX}$		650	
Aux Prop Airfoil selection				NACA 0012	

Simulation Control Variables (reduces processing time)

Gross Weight Minimum	lbs		2000
Gross Weight Maximum	lbs		7000
Airspeed Minimum	kts		50
Airspeed Maximum	kts		300
RFR vs RFA accuracy for optimization			0.01

Figure 5.1: RF_1 Configuration Input Page

Once all of the information was entered into the cells above, the user would press the “Optimize Gross Weight” button and the information shown in Figure 5.2 would be presented.

Calculate Vbr-99 and Vbe for non-optimized GW

Optimize Gross Weight

Simulation Output Parameters - Do not manually change these parameters.

Minimum Gross Weight	lbs	W_G	2429.6875	2917.96875	4226.5625
Power Available 6000ft/95F (MRP)	hp	$PA_{HI/HOT}$	293.4484156	414.985998	999.683685
99% Max Range Airspeed	kts	Vbr-99	118	132	235
Range SFC at Takeoff		SFC_R	0.466483075	0.44502747	0.37565066
Range Power Required at Takeoff	hp	THP	213.6785083	303.621685	1162.89214
Max Endurance Airspeed	kts	Vbe	64	73	1
Total Endurance Power Required	hp	THP	134.2727316	190.652953	486.979588
Empty Weight	lbs	W_E	1336.328125	1750.78125	2747.26563
Weight of Fuel Required for Range	lbs	W_{FRR}	231.4421775	281.538692	501.454333
Weight of Fuel Required for Reserve			114.7293173	155.342378	352.73325
Weight of Fuel Required for Idle and Hover			17.13004721	22.9729724	48.685119
Weight of Total Fuel Required	lbs	W_{FR}	363.301542	459.854043	902.872702
Fuel Weight Ratio Required		R_{FR}	0.149526037	0.15759389	0.21361868
Fuel Weight Ratio Available		R_{FA}	0.110770229	0.1154465	0.14745036

Figure 5.2: RF_1 Output Summary

RF_1 provided an adequate solution for determining trends for gross weight between the three considered design concepts. However, it was missing several features that would be required to finalize the project. It did not use a Combined Blade Element Momentum Theory (CBEM) model, so the performance calculations were a bit too coarse for preliminary design. Also, it assumed an empty weight fraction instead of calculating the exact empty weight. This proved to be a very sensitive parameter that was a bit too important to arbitrarily guess. As a result a much improved spreadsheet was developed which will be referred to as RF_2.

5.2 RF_2 Design Example

The second iteration, RF_2, was developed in February and March of 2007 in order to conduct preliminary design for the 2007 AHS Student Design Competition. Several improvements to RF_1 were made during the development of RF_2. Some of these improvements include:

- A blade element model which allows for blades with nonlinear twist, nonlinear taper, and changing airfoil sections along the span of the blade.
- An empty weight model using Prouty's weight equations and several NASA and military weight models.
- A wing model which allows for different wing and airfoil designs such as box wings and swept anhedrals
- A better intermeshing / coaxial rotor model
- The ability to change rotor tip speeds at different points during the mission.

The user interface for this program became very complex, and it spanned several input pages, so no screen shots of the interface will be presented here. The flow block diagram is shown as Figure 5.3 below.

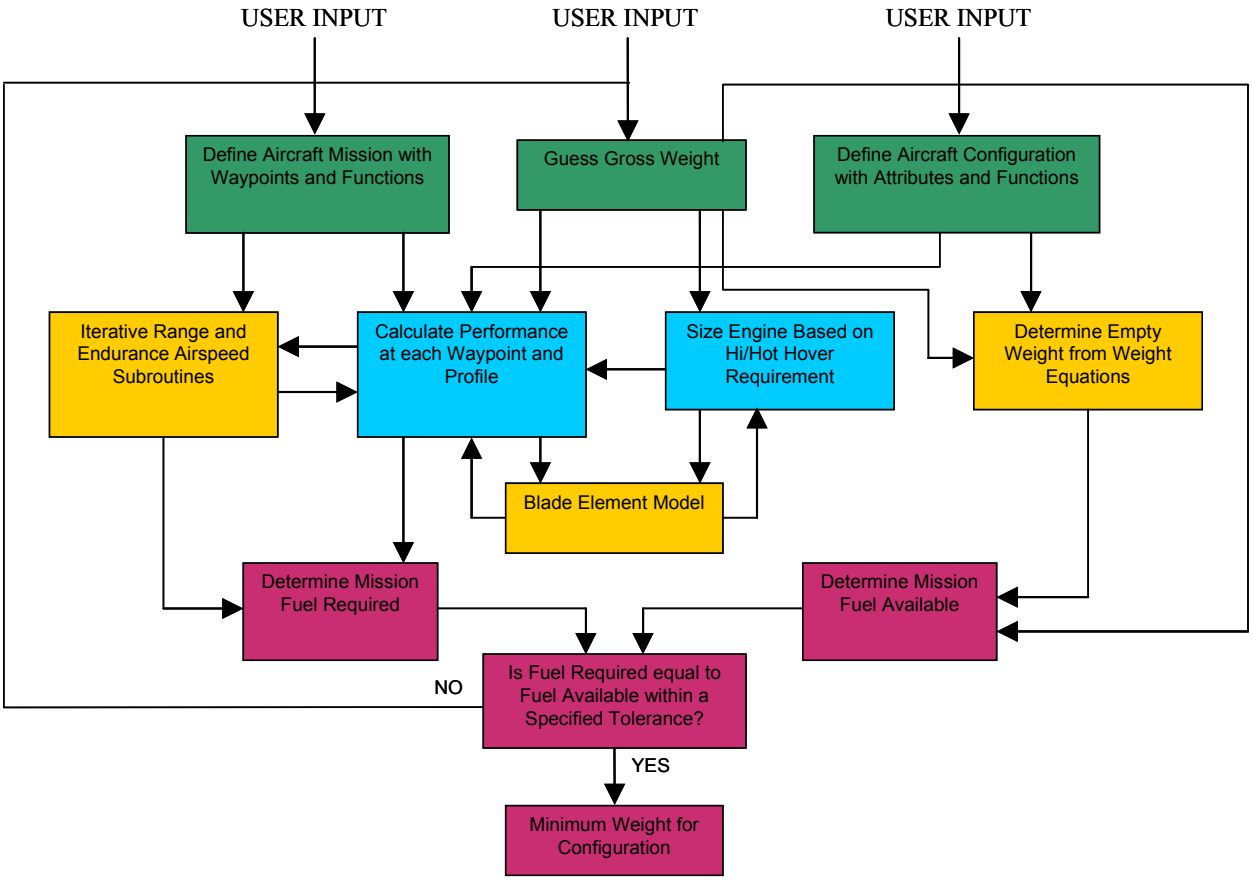


Figure 5.3: RF_2 Flow Algorithm Block Diagram

RF_2 was successful in modeling almost any type of feasible rotorcraft design concept, including the two design solution for the AHS Student Design Competition. A comparison of the rotor model of RF_2 and the TURNS CFD software was conducted, and the results were consistently within 5 percent of each other for both the single main rotor helicopter and the tail sitter concept, so confidence was achieved for the performance algorithms being used. RF_2 became a very useful tool during the design competition, but it was designed for just that purpose. Some consideration was given to other applications of the program, but when time grew short, compromises were made to tailor the program to the specific requirements of the design competition. Also, because it was written in Microsoft Excel with a CBEM, the processing time

was significant. In the case of the box wing tail sitter, a single R_F fuel balance for a given rotor diameter required an average of 30 minutes of processing time.

5.3 Summary of Previous Iterations and Lessons Learned

The developments of RF_1 and RF_2 provided several insights into the design goals of CIRADS. RF_1 has several draw backs in terms of accuracy and modularity. However, it is quite simple to use. A first cut approximation of aircraft sizing can be made with less than 20 design parameter inputs. This made the program usable by almost anyone on the Georgia Tech rotorcraft design team. Also, the processing time is not very long. All three concepts (single main rotor helicopter, coaxial helicopter, and tilt rotor) are sized in under a minute under most circumstances.

RF_2 has a significant level of accuracy and modularity, but towards the end of its development, it became extremely cumbersome. Very few people are proficient at using it. Also, the processing time is significant. It would not be very attractive for a user wanting to see general trends or a quick estimate of a configuration on a purely conceptual scale. The program requires multiple inputs for detailed physical aircraft parameters as well as weight component breakdowns. No options are available to choose a less detailed method of calculation. So, while the capability of RF_2 is impressive, it has fallen into the trap of several other software tools used in industry today. It is too complicated for anyone to use.

5.4 CIRADS Developmental Goals and Considerations

Based on the lessons learned in developing RF_1 and RF_2, three overall high level goals for a third generation development had been decided.

1. CIRADS should have the capability of modeling almost any type of feasible rotorcraft configuration to include:
    a. Single Main Rotor Helicopters
    b. Coaxial Helicopters
    c. Tandem Helicopters (with varying degrees of intermeshing rotors)
    d. Tilt Rotors
    e. Compound Helicopters
    f. Tail Sitters
2. CIRADS should be highly configurable to allow different levels of detail and accuracy for different levels of conceptual and preliminary design. This includes being able to model aircraft with general efficiencies and figure of merit values with assumed empty weight fractions to modeling aircraft using CBEM codes with complex airfoils and detailed component weight breakdowns to modeling aircraft with various levels of detail in between these two extremes.
3. CIRADS should have a very intuitive graphical user interface which allows user of various experience levels to use the software with minimal training.

These three goals are almost as conflicting as any three goals could be. This is evident in many commercial software applications. The desire for maximum capability often leads to applications which are very difficult to use and require significant formal training and experience. Also, the first goal implies breadth whereas the second implies depth. Balancing these three goals requires several compromises and considerable engineering judgment.

5.4.1 Object Oriented Applications

The nature of rotorcraft analysis and design lends itself to an object oriented environment. An aircraft is made up of multiple components (i.e. rotors, engines, fuselage, wings, tail rotor, etc). These components may be considered as objects which may be used on other aircraft. Also, the rotor itself has subcomponent features (i.e. diameter, solidity, airfoil type, number of blades, etc). It would be convenient to be able to apply features, such as airfoil type, to a different rotor. An airfoil type also has subcomponent features (i.e. lift curve slope, drag polar equation, drag divergence information, stall angle of attack, etc). Also, in the case of design, the aircraft itself is a subcomponent of the overall design project. The mission that the aircraft must fly would also be a subcomponent of the overall design project. It would be convenient to apply a different aircraft to the project to see how well it performs the specified mission of that design project. So, in essence, rotorcraft analysis and design are object oriented problems that require object oriented solutions.

Each of the components (or objects) of a project should be able to be saved in such a way that they can be archived and applied to other projects. It would be overly demanding on a user to have to specify everything about every component of a project for each project. It would also be overly cumbersome to expect a user to have to have a separate copy of the entire software package for each project. This was the case with RF_2. The Excel spreadsheet could only handle one aircraft and one mission at a time. During the 2007 AHS Student Design Competition, there was a separate spreadsheet for the manned and the unmanned aircraft. In this sense, RF_2 was not an application. It was just a spreadsheet. CIRADS should have the ability to access libraries of objects, and it should be able to save and open these objects one at a time or as a subcomponent of a larger component in the hierarchy. For example, the user should be able

to input engine data and save it as a file which can be opened later for viewing and editing. Then, the user should be able to specify an aircraft configuration and be able to choose the engine file that was saved as a component of the aircraft (possibly by choosing it from a combo box). An entire design project should be one file which can be saved and opened later for viewing and editing. Within that file, should be a link to other files (i.e. Mission and Aircraft). Those files should also have links to other files (i.e. the aircraft file should have a link to the type of engine used). All of this should be done within the graphical user interface of one copy of the software. Therefore, CIRADS should be more than just a program. It should be an application.

5.4.2 Software Development Environment Selection

A major consideration prior to beginning a software project is evaluating the software tools available and recommended for developing the project. During software selection, the developer should consider the following.

1. the capability of the software to meet all of the design goals
2. difficulty level of the development environment
3. time required to finish the project
4. developer experience level with possible software tools
5. processing speed of the finished code
6. availability for use on multiple platforms (i.e. Windows, Mac, etc)
7. future modifications and experience levels of those most likely to perform the modification

5.4.2.1 Microsoft Excel

RF_1 and RF_2 (the predecessors of CIRADS) were both written in Microsoft Excel. Excel provides a very good development environment for small programs. Iterative calculations can be performed within the spreadsheet cells with no initialization (the iterative calculations box must be checked in the options window). Also, a simple graphical user interface can be designed in Excel quite easily. For advanced programming, the Visual Basic for Applications (VBA) programming modules within Excel provide all of the necessary capability required for a program like CIRADS. However, the processing speeds of code developed in Excel are quite slow compared to other environments such as C, Java, FORTRAN, or MATLAB. Also, Excel does not lend itself well to the objected oriented environment described in the previous section. There is a feature for building graphical forms in Excel, but the capability is limited. Also, when a spreadsheet becomes very large and complicated, it is quite difficult to debug, because the code is usually not in one place. The programmer must go from cell to cell trying to find the problem. Due to these set backs, Microsoft Excel was not chosen for CIRADS development.

5.4.2.2 C++ and Visual Basic

Both C++ and Visual Basic have extensive graphical user interface development capability and both are capable of meeting all of the programming goals for developing CIRADS. In fact C++ provides exactly the type object oriented programming environment for developing an application like CIRADS. The major disadvantages of C++ and Visual Basic are their inherent attachment to only the Windows Operating System and the lack of C++ programming experience within the Georgia Tech Aerospace Department. As a result, C++ was strongly considered, but not selected for CIRADS development.

5.4.1.3 Java

Java also provides an extensive graphical user interface development capability and object oriented programming environment. Also, Java is capable of running on virtually all operating systems with minor code linking changes when switching from one operating system to another. Almost all computers are shipped with a Java Runtime Environment preinstalled. Additionally, there are members in the Georgia Tech Aerospace engineering department familiar with Java. Java's drawbacks are that it is not extremely fast in terms of processing speed, and it is not very convenient for conducting complex calculations like those described in Chapters 3 and 4. Therefore, Java was chosen for the graphical user interface development software, but it was not chosen to be the developmental software for the iterative analysis and design software algorithms using the equations in Chapters 3 and 4.

5.4.1.4 MATLAB

Almost all Aerospace Engineering students and faculty at Georgia Tech are at least familiar with MATLAB, and many of them are very proficient. MATLAB is shareware and not considerably inexpensive if purchased outside of an educational environment. However, MATLAB can be compiled into a stand alone executable that can be run on a computer that does not have MATLAB installed. So, from the perspective of a user, no special software is required to run a graphical user interface written in Java linked to a stand alone executable. MATLAB also provides a very user friendly programming environment with easy to learn syntax. In fact, later versions of MATLAB are capable of designing graphical user interfaces that also may be complied as stand alone executables. So, the entire CIRADS program could have been written using just MATLAB. However, experience with developing graphical user interfaces using

MATLAB is somewhat limited at Georgia Tech, and it is not quite as well documented with user blogs on the internet to the degree that Java is. Therefore, MATLAB was not chosen to be used for developing the graphical user interface, but it was selected for developing the iterative analysis and design algorithms using the equations in Chapters 3 and 4.

5.4.3 Graphical User Interface Development

During the development of CIRADS, the development of the graphical user interface (GUI) spearheaded the design effort. It was 85 percent completed before any processing algorithms were started. This has often proven to be a best practice in developing software. Considering the use of the software from the perspective of the end user prior to getting bogged down in the details of complex algorithms gives the software developer a more objective perspective. It is also a good practice to have peers not involved in the development of the software (or focus groups) evaluate the user interface prior to beginning detailed code development. The GUI will be briefly presented in the next chapter for the purpose of illustrating design theory. For a detailed description of the GUI and instructions for using CIRADS, refer to the CIRADS user's manual. The primary purpose of this thesis is to document the design practices and considerations for developing the software. So, illustrations will be given in the next chapter, but not to the degree to teach the software.

CHAPTER 6: CIRADS OVERVIEW

All of the design goals mentioned in the previous chapter were successfully completed with the exception of completing the CBEM. It was not completely integrated due to time constraints. The two methods available for calculating power required are 1) based on efficiency factors and figures of merit, and 2) based on a modified momentum theory model with drag polar equations and empirical correction factors as described in Chapter 3. However, through testing, the accuracy of this model has been very surprising. The equations were calibrated against the Hiller 1100 and the XV-15, so those aircraft model almost perfectly. Trial runs for other configurations have shown remarkable accuracy definitely appropriate for conceptual design. Also, the graphical user interface was developed for future integration of a CBEM. A rotor airfoil design module was created with a rotor CBEM testing platform. The aircraft configuration pages are equipped with options for selecting CBEM based rotors, but they are not yet active. So, when a CBEM has been developed and properly tested, it should be integrated successfully with minimal required GUI changes.

6.1 CIRADS Architecture

The general architecture of CIRADS is shown in Figure 6.1 below. At the highest level of the architecture are CIRADS.jar, CIRADS.exe, and the lib folder. CIRADS.jar is the Java GUI. Double clicking on it will launch CIRADS. CIRADS.exe is the program developed in MATLAB but compiled as a stand alone executable for processing the iterative analysis and design equations presented in Chapters 3 and 4. Double clicking on it will have no useful effect without intricate knowledge of text file exchanges within the CIRADS architecture.

CIRADS.exe is intended to be launched from within the GUI for general users, so double clicking on it is not recommended. (Note – it will not harm anything if a user double clicks it. It just will not do anything useful without other text file features setup artificially. For information on how to do that, see the CIRADS user's manual. Otherwise, CIRADS.exe is launched normally from within the GUI when necessary).

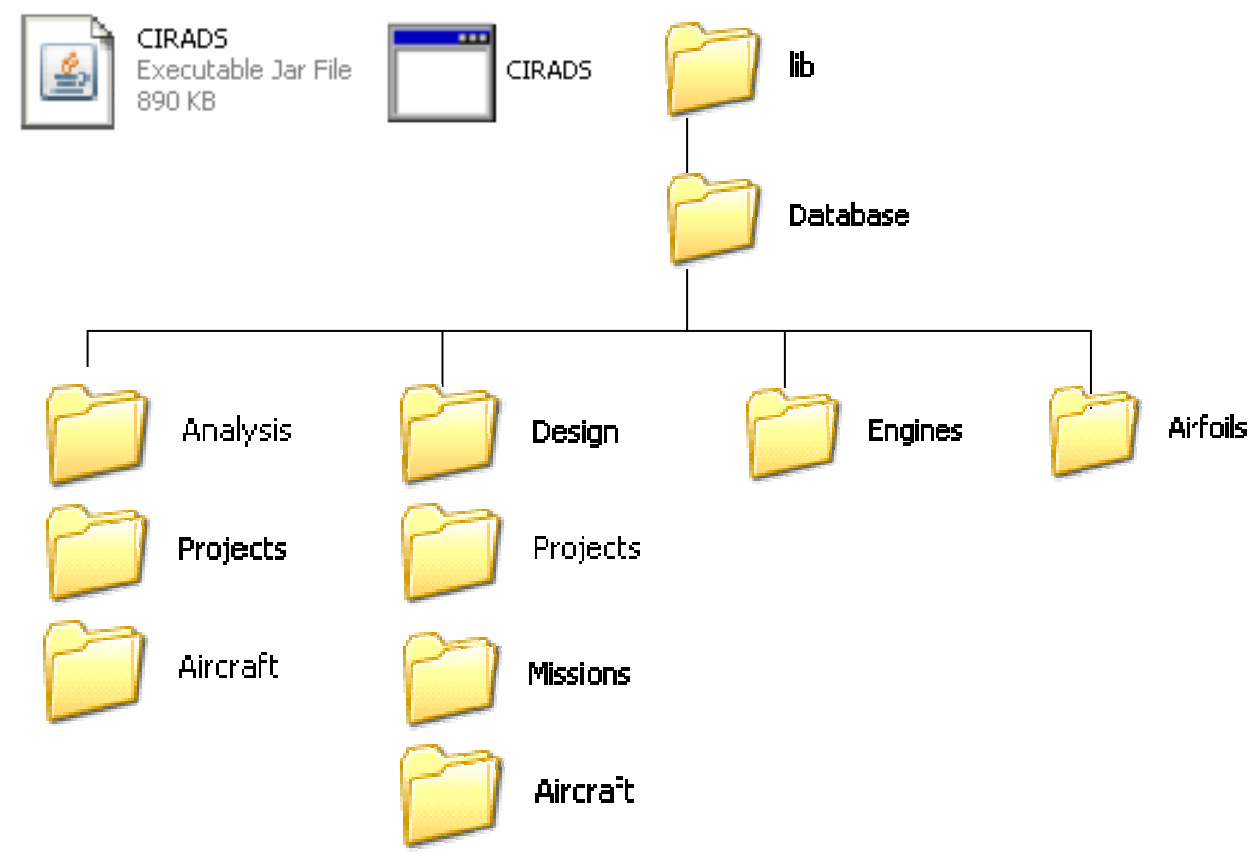


Figure 6.1: CIRADS Architecture

The "lib" folder is a folder created automatically by Java during compile time. It is intended to store libraries such as the swing.jar library which Java references for certain objects such as text fields, combo boxes, labels, etc. This folder is also used to store the "Database"

folder which contains the architecture for all of the data input files and folders necessary for CIRADS to operate. The files stored within all of the folders subordinate to the "Database" folder are simple text files with all of the necessary object oriented data discussed in the previous chapter. The four folders in the "Database" folder are "Analysis", "Design", Engines", and "Airfoils".

The reasons for this architecture will become clearer after the GUI has been introduced. For now, "Analysis" is used to find the full performance envelope of an aircraft that has already been designed. "Design" is used to size a new undefined aircraft based on a set of mission requirements. For example, if a user wanted to see all of the performance parameters of a UH-60, the user would use the analysis module. If a user wanted to size an aircraft based on the mission requirements for the 2008 AHS Student Design Competition, the user would use the "Design" module. Then, if the user wanted to see the full performance of the newly designed aircraft for the 2008 AHS Student Design Competition, he would use the analysis module. As might be expected, the "Design" module is more complex than the "Analysis" module in terms of user input. Distinguishing between analysis and design is very appropriate. It makes the user interface much more intuitive, and it allows the user to focus on one piece at a time.

The "Engines" and "Airfoils" folders are at the same echelon as "Analysis" and "Design" in the overall directory architecture, because they are common to both modules. An engine may be used in an aircraft in the "Analysis" module, and the same engine (scalable or not) may be used in an aircraft in the "Design" module. Therefore, for the purpose of directory hierarchy, "Engines" and "Airfoils" are at the same level as "Analysis" and "Design". This architecture should not be confused with the file linking hierarchy shown in Figures 6.2 and 6.3 below.

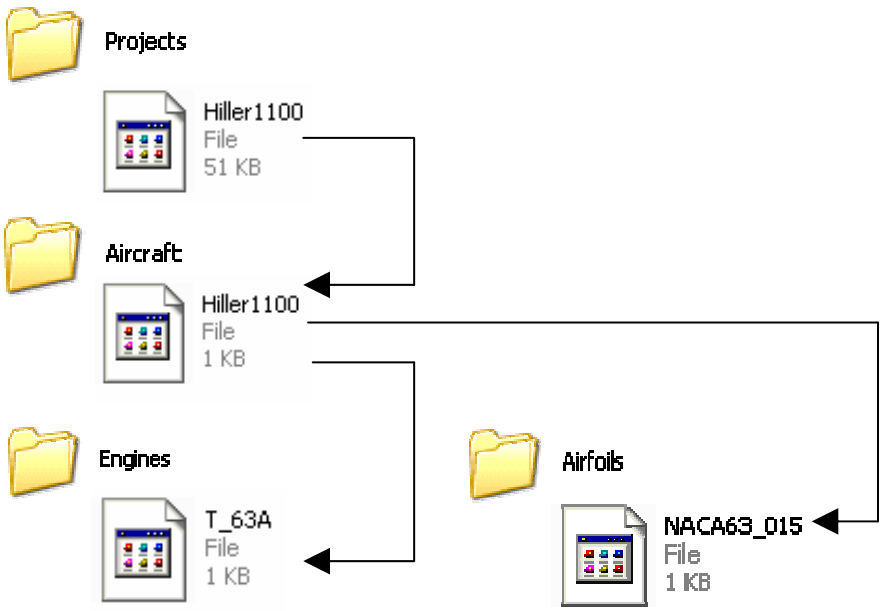


Figure 6.2: CIRADS Analysis File Linking

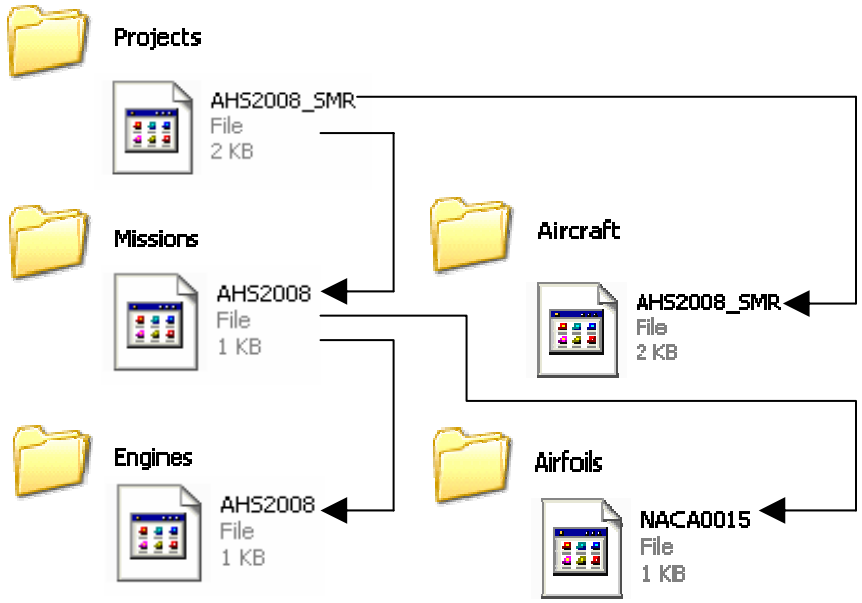


Figure 6.3: CIRADS Design File Linking

For the purpose of configuration hierarchy, it should be obvious that an engine is a subcomponent of an aircraft, which is a subcomponent of a project. So, one of the lines in a "Project" file is the name of the "Aircraft" file, and one of the lines in the "Aircraft" file is the name of the "Engine" file. During runtime, CIRADS will go to those appropriate directories and retrieve the appropriate file.

The highest level of the file structure is the "Project" file. In the "Analysis" module, the "Project" file contains three basic things.

1. The name of the "Aircraft" to perform analysis on (a reference to the file)
2. The type of analysis to perform (i.e. hover power vs. hover height, power required vs. airspeed, max airspeed vs. altitude, max airspeed vs. gross weight) and the variable step sizes. (all entered on "Setup" tab)
3. The output of the iteration (displayed on "Output" tab)

For the design module, the format is the same, but with the addition of the "Mission" name (a reference to the file).

6.2 Graphical User Interface

The overall layout of the graphical user interface (GUI) is a tabbed pane hierarchy that resembles that of the CIRADS architecture. After double clicking on the CIRADS.jar file, the GUI will launch and the "Main" page of CIRADS will be displayed as shown in Figure 6.4.

Figure 6.4: CIRADS MAIN

Notice that the tabs at the top of the GUI match the architecture shown in Figure 6.1.

6.2.1 Engine Module

An example screen shot of the "Engine" Module is shown in Figure 6.5 below. Notice that the user has the option of choosing a default engine or one that is based on actual engine data. Not seen on this screen shot are sections to specify equations for SFC correction factors with engine scaling and SFC and HP corrections with RPM variation. If the user selects the "Default Turboshaft Engine Definition" option, the MCP at sea level ISA and associated SFC are required inputs. The degradation of power required vs. altitude and temperature will use the default constants mentioned in Chapter 3. The same is true for SFC degradation with partial power and

temperature. If the "Advanced Engine Definition" option is selected, the engine constants will be determined iterative from the actual data supplied.

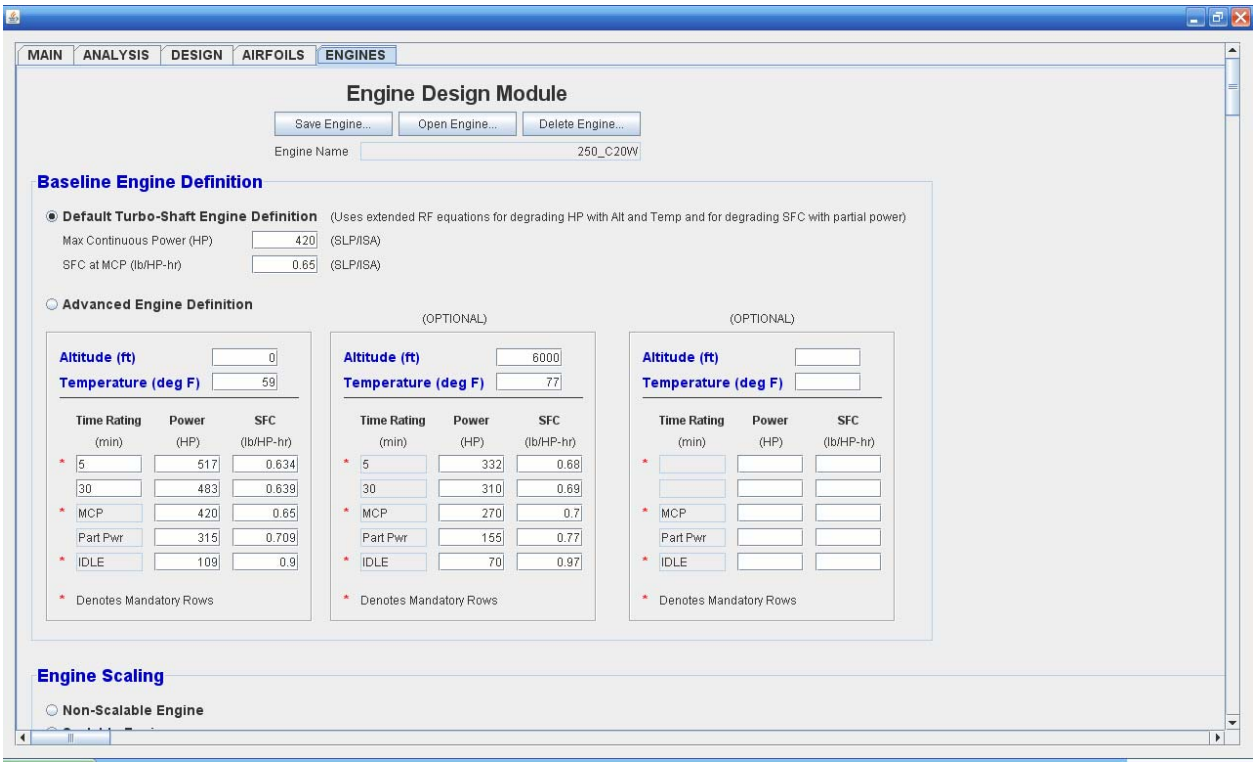


Figure 6.5: Engine Design Module

Once a user has specified all of the required engine information, the "Save Engine…" button may be pressed to save the engine file and a save dialog box will appear.

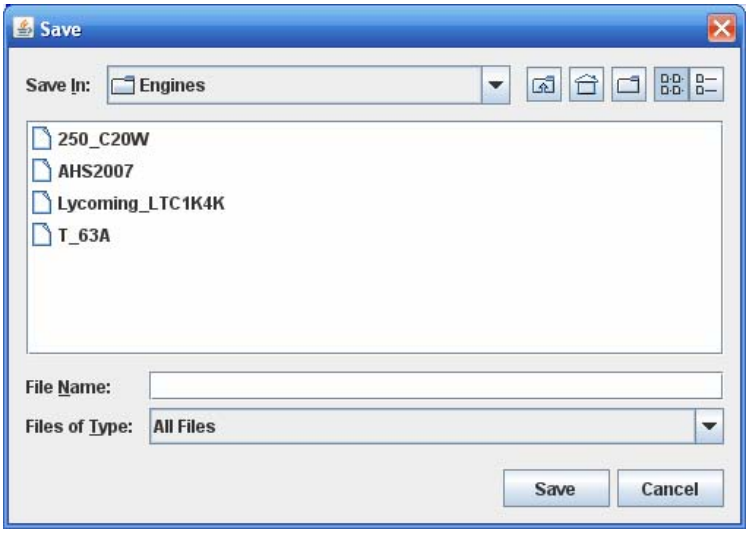


Figure 6.6: Engine Save Dialog Box

6.2.2 Airfoil and Rotor Blade Design Module

Figure 6.7 shows the "Airfoil and Rotor Blade Design Module". Here, the user may specify airfoil data. Under the current CIRADS capability, only drag polar airfoil definitions may be used with the information shown in the figure. However, the C81 Based Airfoil Definition (CBEM) GUI has been completed. It is not completely shown in the screen shot. The "Save Airfoil…", "Open Airfoil…", and "Delete Airfoil…" buttons all work the same way as shown for the "Engine Design Module". This way, the user is never required to navigate through the architecture folders.

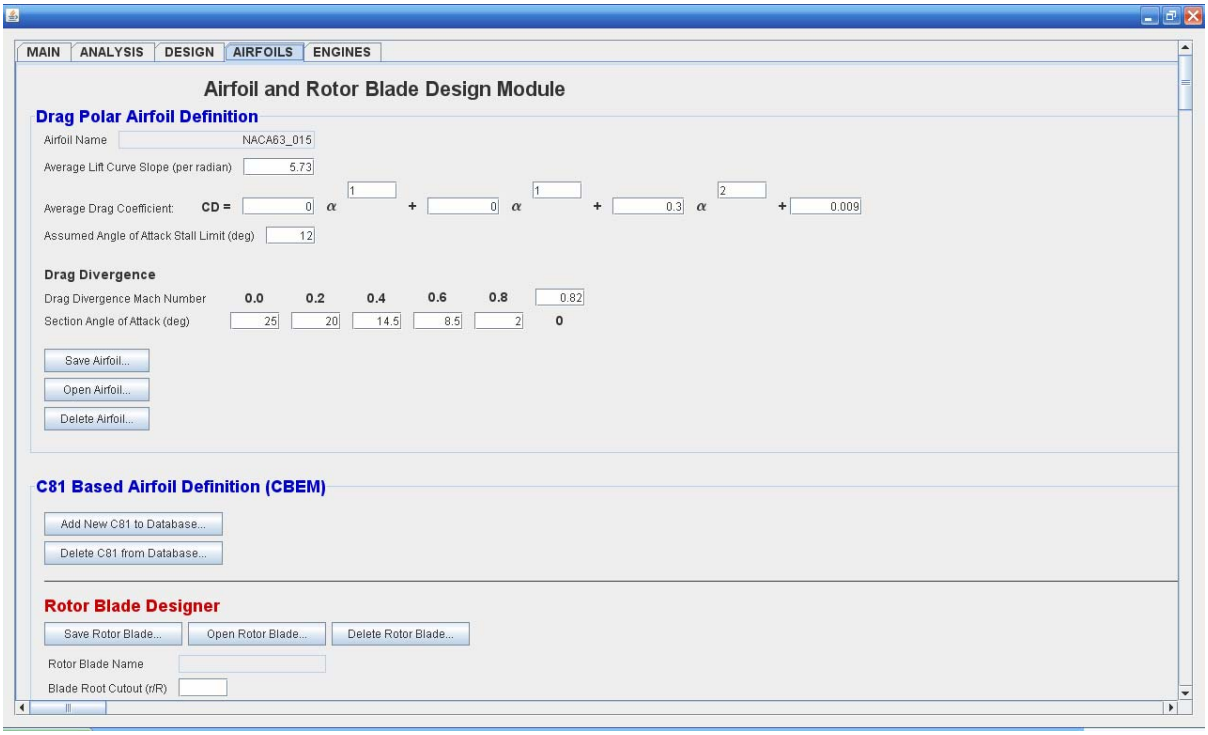


Figure 6.6: Airfoil and Rotor Blade Design Module

6.2.3 Analysis Module

The "Aircraft" configuration tab for the analysis module is shown in Figure 6.7 below. Most of this tab is not shown in the figure due to size. There are several sections for specifying components such as main rotor, engine and fuselage, anti-torque, wing, and auxiliary propulsion. Notice, that there are two general vertical columns denoted with blue labels at the top. The column to the left is the most basic information needed to define an aircraft. Most of the fields in this column for all section are based upon general efficiency factors and figures of merit. The second column allows for more detail. Radio buttons are used to distinguish between which method is being used. This way, the user has the freedom to specify as much or as little detail as desired. Also notice that toward the bottom of the figure is a combo box for selecting the type of airfoil drag polar. The airfoils displayed in the combo box represent the files that are saved in

the “Airfoil” folder shown in Figure 6.1. An option is available for specifying a custom rotor blade using a CBEM, but this method has not yet been implemented.

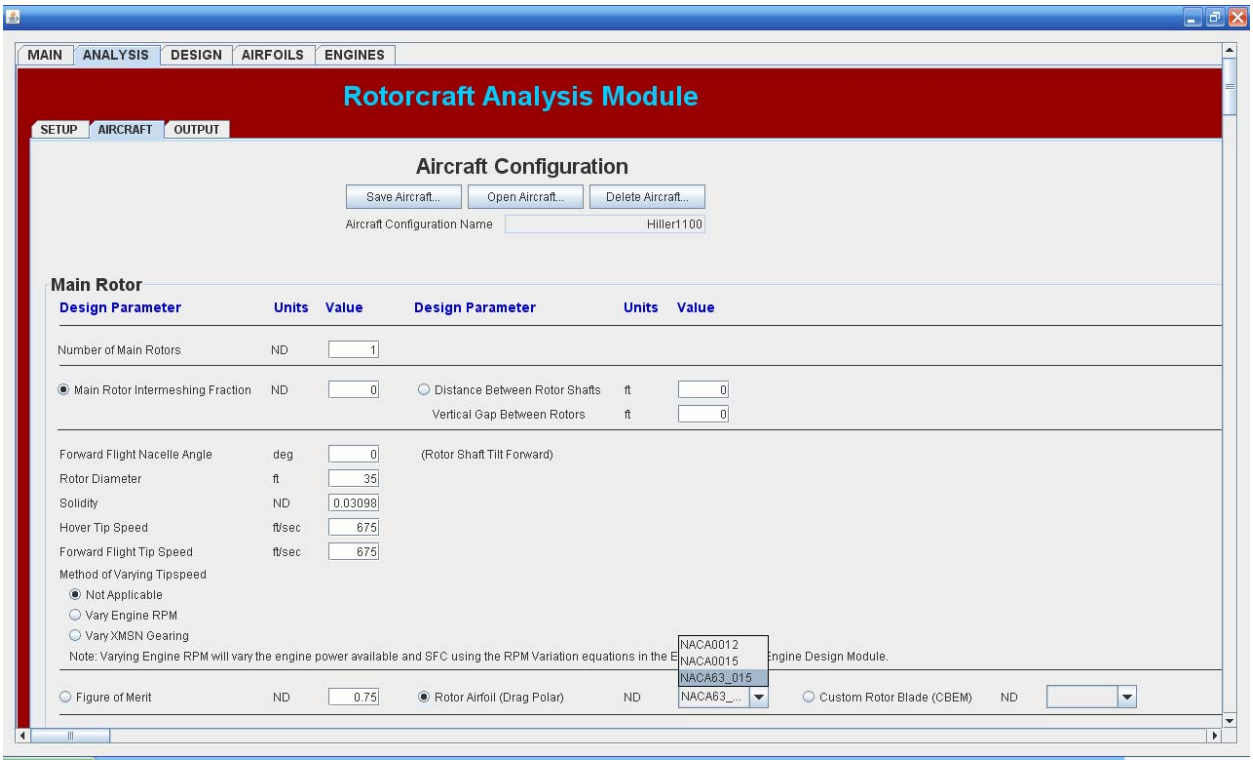


Figure 6.7: Analysis Aircraft Configuration Module

The setup tab for the “Analysis” Module is shown in Figure 6.7 below. Notice that just under the “Run Simulation” button is a combo box for selecting the “Aircraft”. This combo box contains every file that is saved in the “Aircraft” folder in the “Analysis” folder. Also notice that below the aircraft combo box are sections for specifying the type of analysis to conduct. If the check box at the top of each section is checked, the analysis will be performed. Clicking the “Run Simulation” button will begin the simulation. During this time a command window will open. This shows that the CIRADS.exe has been called. Upon completion of the simulation the command window will close, and the output from the simulation will be presented on the

"Output" tab as shown in Figure 6.9. Most of the tables are not shown in the figure, but they display a very wide variety calculated performance parameters.

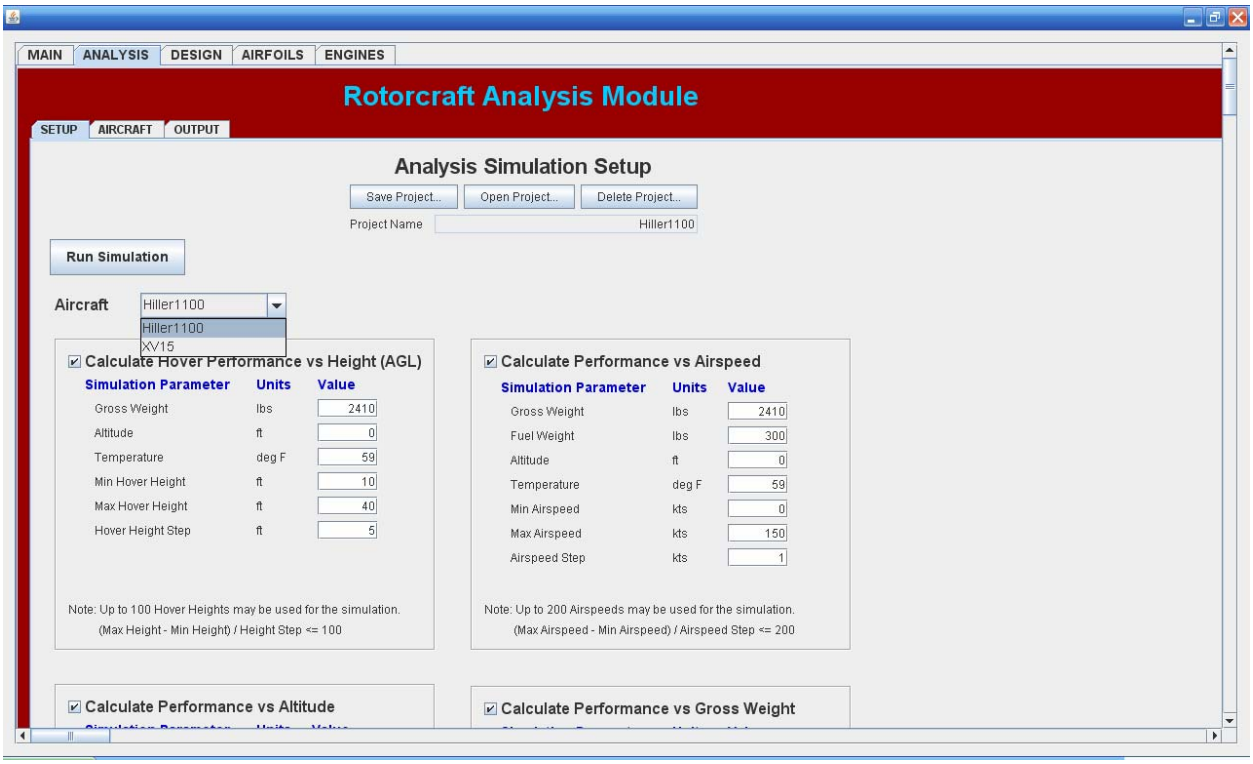


Figure 6.8: Analysis Simulation Setup

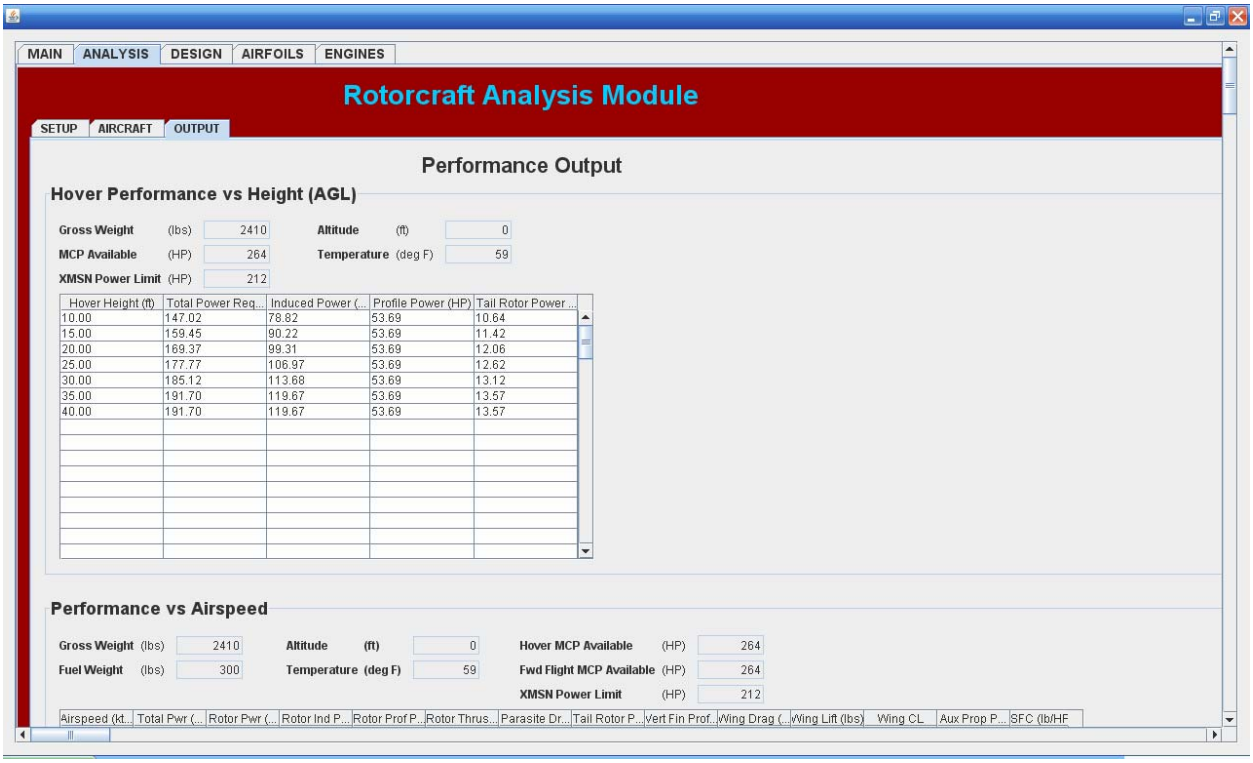


Figure 6.9: Analysis Performance Output

In order to conduct an analysis simulation of a UH-60A helicopter, the following steps would need to occur:

1. Use the “Engines” input module and define the T-700 engine using the fields required on that page and save that engine file.
2. Use the “Airfoils” input module and define the UH-60 airfoils (SC1094 and SC1095) using the fields required on that page, and save the airfoil files.
3. Go to the “Analysis” module, select the “Aircraft” tab, and enter all of the necessary data for the UH-60 (choosing the T-700 and SC1094/SC1095 airfoil from combo boxes on that page) and save the aircraft file.

4. Go to the setup tab, select the desired aircraft to perform analysis on (the recently saved UH-60 file) and specify the type of performance information desired using the fields on the setup page.
5. Press the "Run Simulation" button. Once the file has completed processing, go to the output tab, and all of the analysis data will be displayed.

6.2.4 Aircraft Design Module

The "Design Simulation Setup" tab is shown in Figure 6.10 below. Notice that there are now two combo boxes below the simulation button. One is for the mission, and the other is for the aircraft. These are linked to the appropriate directories of Figure 6.1.

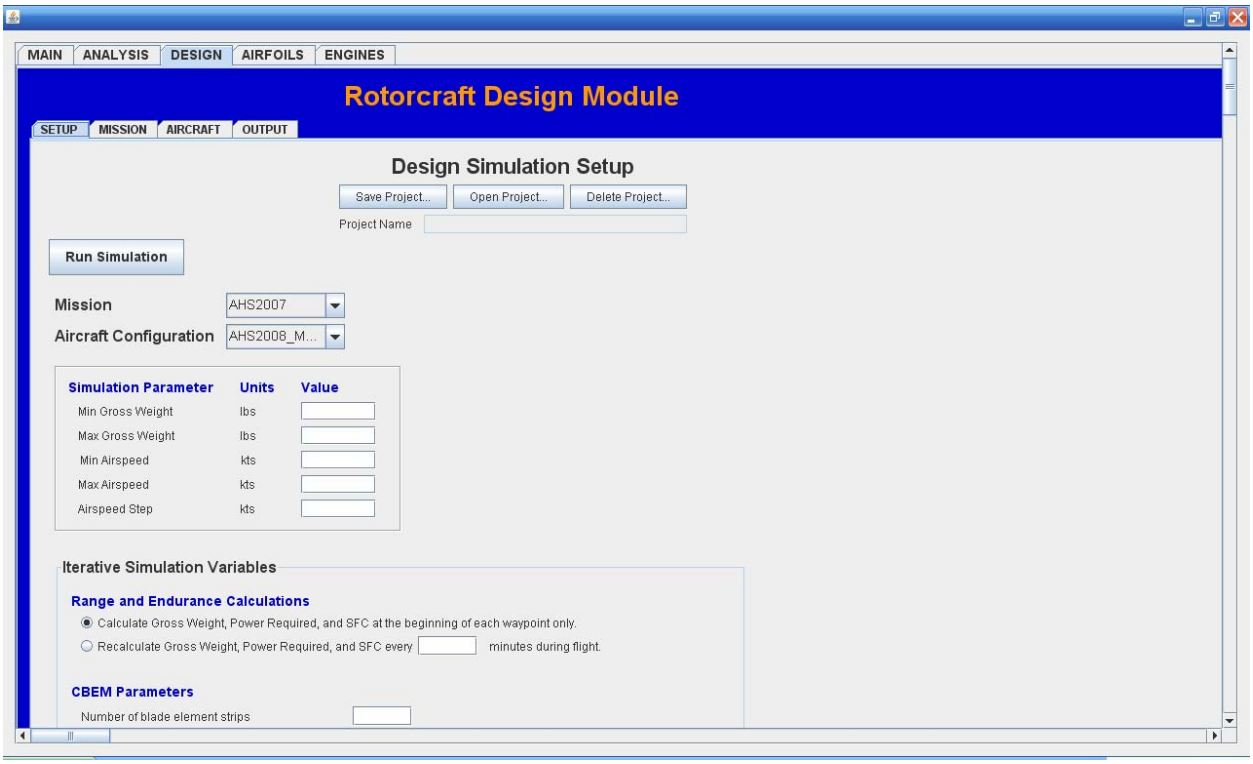


Figure 6.10: Design Simulation Setup

Notice that there are not nearly as many setup parameters for the design module as there are for the analysis module. This goes back to the difference between analysis and design. The purpose of the design module is to size the aircraft that best completes the design mission. So, the design module focuses on the "Mission". All other ancillary calculations are not conducted. Those may be performed in the analysis module. Besides, as shown in the figures that follow, the design module inputs for the other tabs get busy enough that other ancillary calculations are neither wanted nor required. Figure 6.11 shows the "Engine Sizing Requirements" section of the "Mission and Sizing Requirements" tab. One or all of the requirements may be given. The engine is sized based on the one that requires the most installed power.

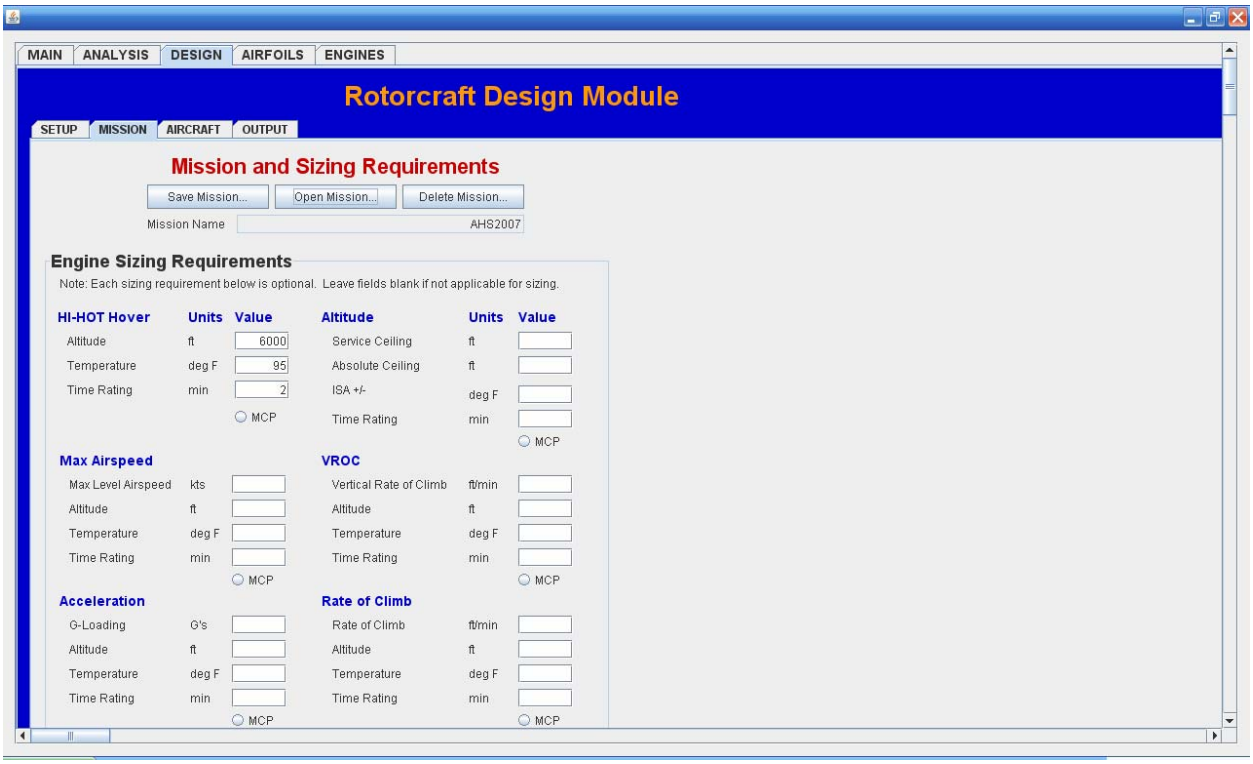


Figure 6.11: Engine Sizing Requirements

The “Sizing Mission” for the “Mission and Sizing Requirements” tab is shown in Figure 6.12 below. This mission shown is the one provided in the 2007 AHS Student Design Competition. Notice that buttons are available to add and delete waypoints. Up to 10 waypoints may be used to specify a sizing mission.

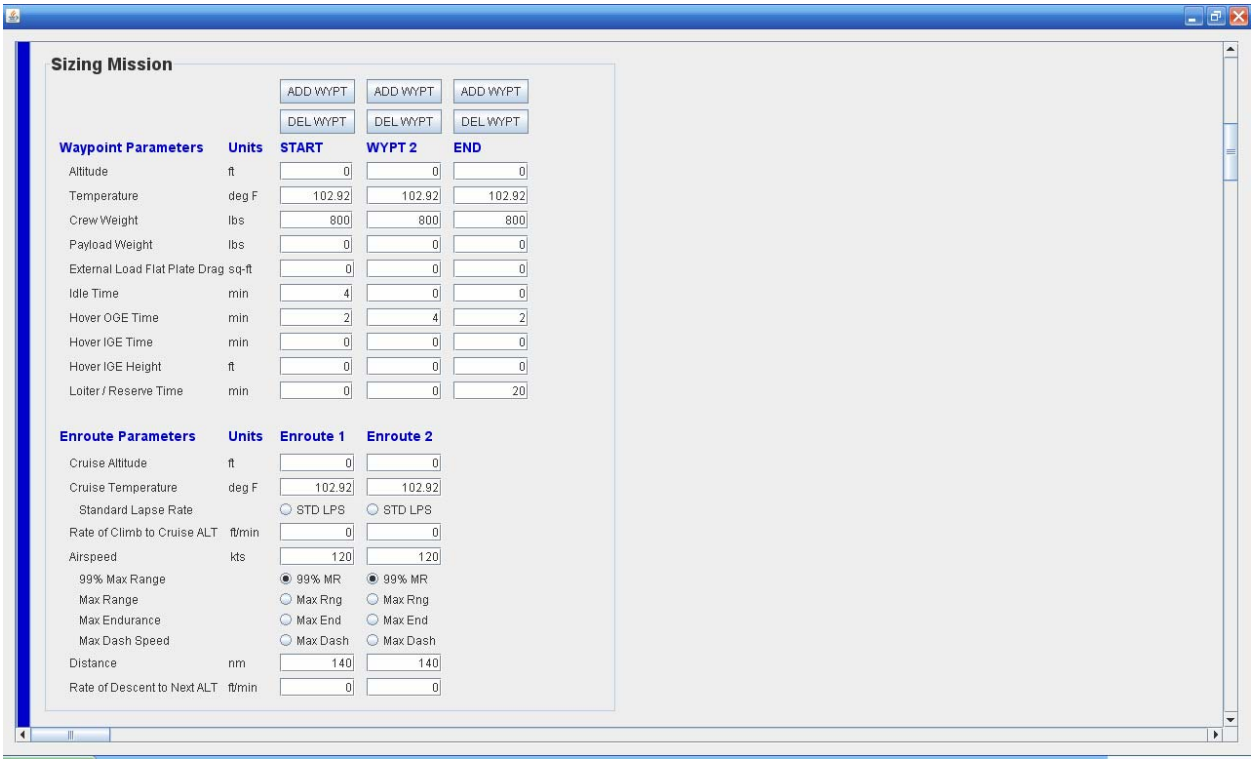


Figure 6.12: Sizing Mission

The “Aircraft Configuration” tab for the Design module is shown in Figure 6.13 below. Notice that this page is very similar to the “Aircraft” tab in the Analysis Module. However, some of the parameters have a checkbox associated with them that allows for varying the parameter during flight. These checkboxes allow for changing the profile of the aircraft at each mission waypoint along the mission. For example, a tilt rotor may be in helicopter mode at one waypoint and airplane mode at another. The checkboxes also allow for rotor morphing during flight. Several

parameters such as rotor diameter, solidity, airfoil type, and tip speed may be changed at each waypoint and between waypoints.

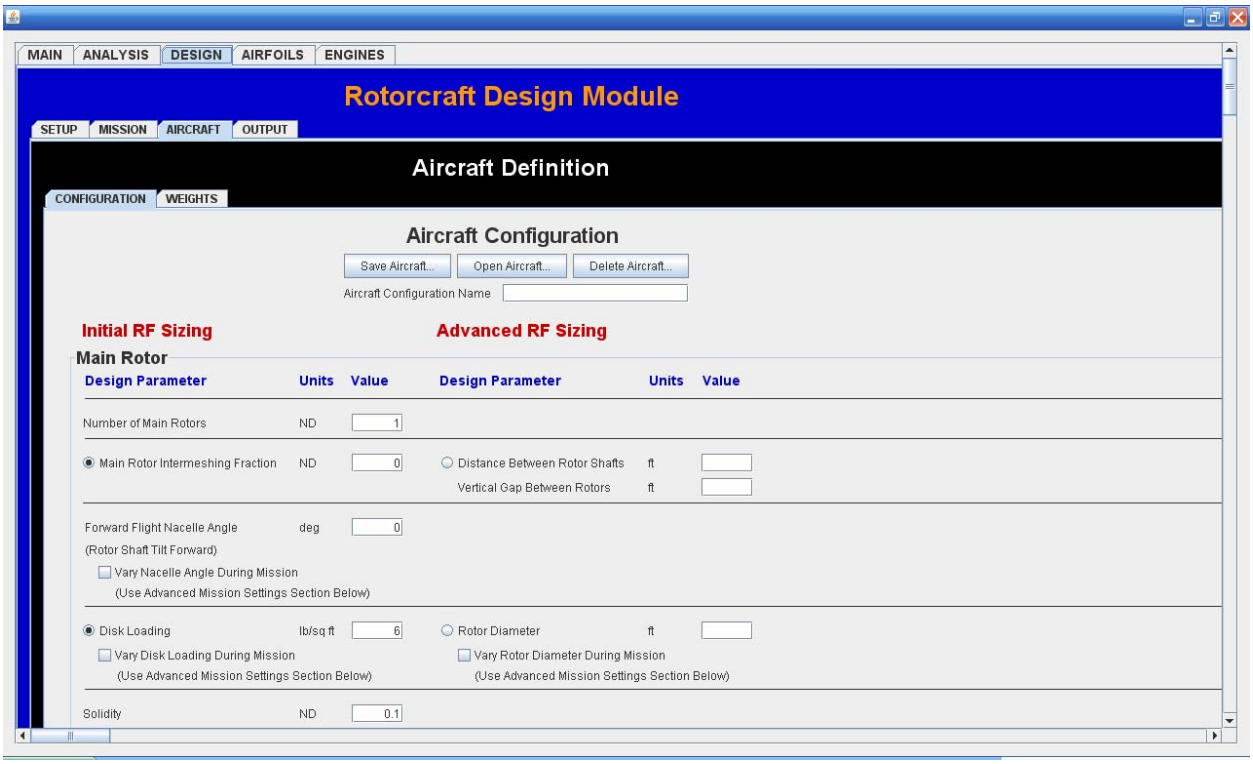


Figure 6.13: Design Aircraft Configuration Tab

Also added to the bottom of the Design Aircraft Configuration tab is an "Advanced Mission Settings" section as shown in Figure 6.14. For all of the parameters that have a check box beside them, the "Advanced Mission Settings" section may be used to specify these parameters at each waypoint. This allows for rotor morphing as well as flight profile changes. The "Advanced Mission Settings" section has virtual visibility of the "Mission" tab. So, the same number of waypoints will be visible in the "Advanced Mission Settings" section as is used on the "Mission" tab. If another waypoint is added to the "Mission" tab, another one will appear in the "Advanced Mission Settings" section.

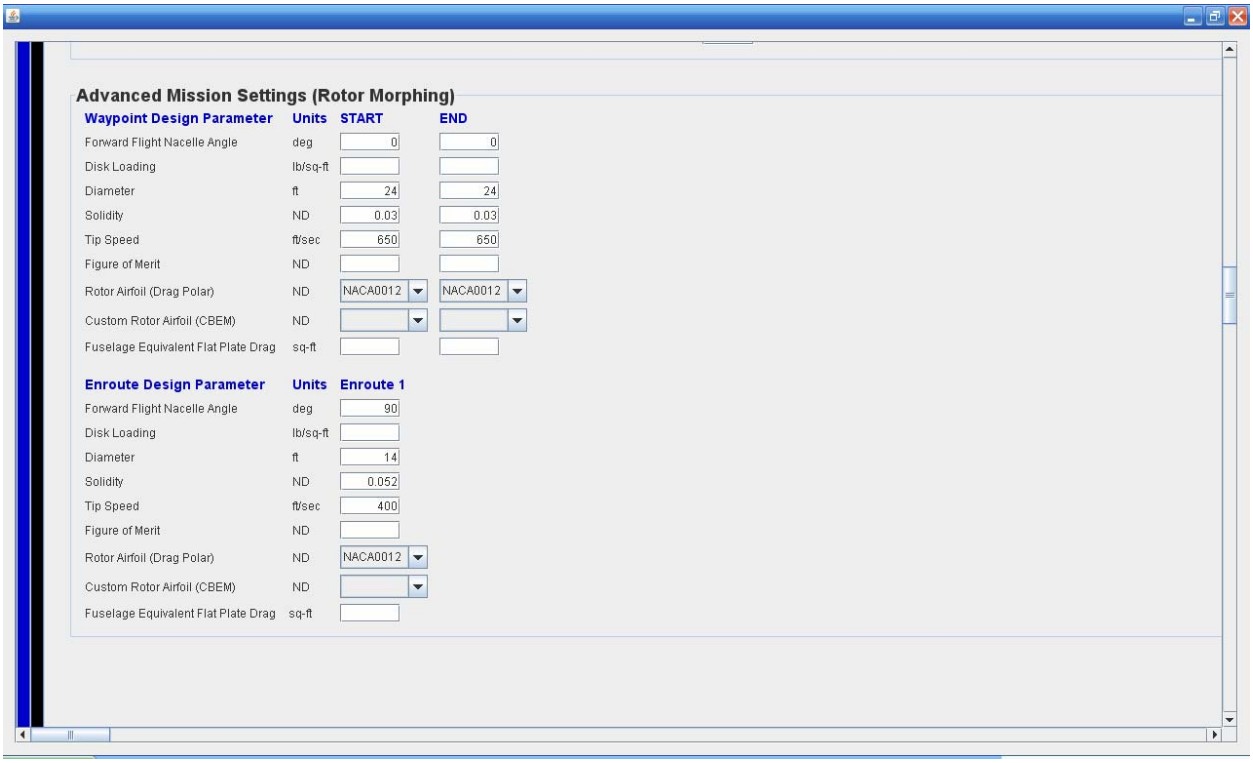


Figure 6.14: Advanced Mission Settings Section

Also notice from Figure 6.13 that a "Weights" tab appears next to the "Configuration" tab. This did not appear in the Analysis Module because it is not necessary to calculate empty weight when analyzing an aircraft. The aircraft has already been designed, so the empty weight should already be known. However, in the Design Module, a major parameter to be determined is the empty weight of the aircraft. As mentioned in Chapter 4, there are two primary methods for determining empty weight when using the R_F method. The first assumes an empty weight fraction. That method is available in the Design Module as well. The second uses empirical iterative weight component calculations to estimate an empty weight for each iteration of gross weight. When the second method is desired, the "Weights" tab in the "Aircraft Definition" panel will be used. Figure 6.15 shows a screen shot of the "Weights" tab. This tab is very extensive so only part of it is shown here. Notice that for each component weight, there is a checkbox for

specifying an iterative equation. Most of these equations are taken from Prouty's *Helicopter Performance and Aerodymanics* [1]. For some components, others equations are available as well. Also notice that for each component, there is a "Tech Factor" and a "Bias". These are used to make manual corrections to the weight components.

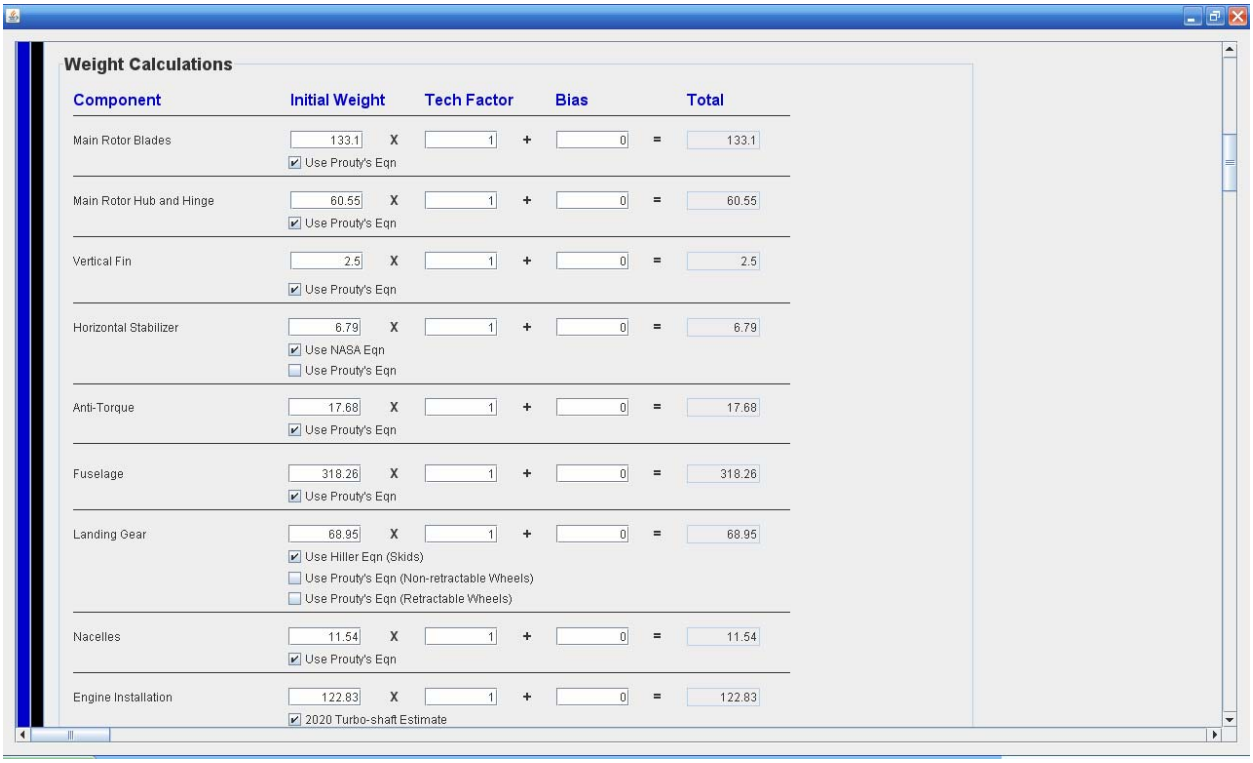


Figure 6.15: Weight Calculations

Finally, the Design "Output" tab is shown in Figures 6.16 and 6.17. Figure 6.16 shows the overall design summary and part of the engine sizing summary. Each of the engine sizing requirements listed on the "Mission" page is listed again here with the power required for that maneuver at the specified environmental conditions as well as the power required reflected back to sea level ISA at 100 percent engine RPM.

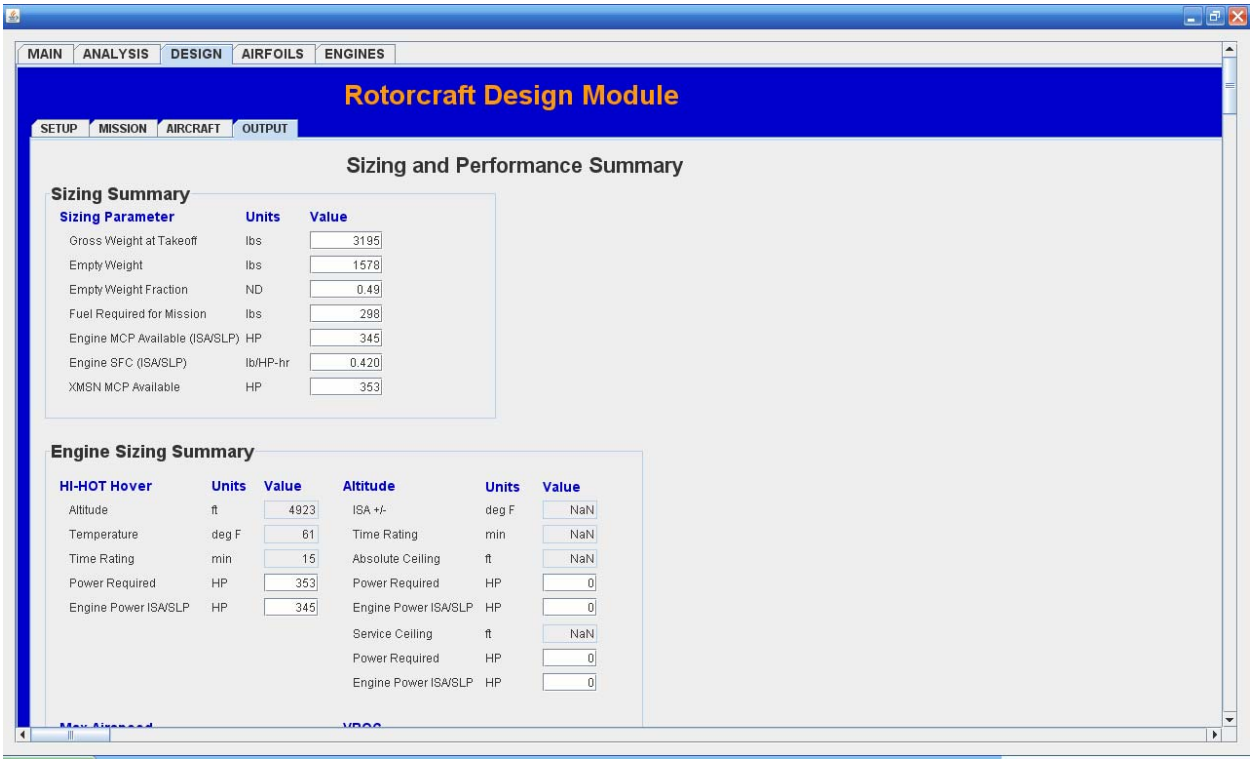


Figure 6.16: Sizing and Performance Summary (Top)

Figure 6.17 shows part of the summary of the sizing mission. Key performance parameters at each waypoint and enroute segment are summarized. The "Output" section is also dynamically linked to the "Mission" tab so the number of waypoints visible here will match the number visible on the "Mission" tab.

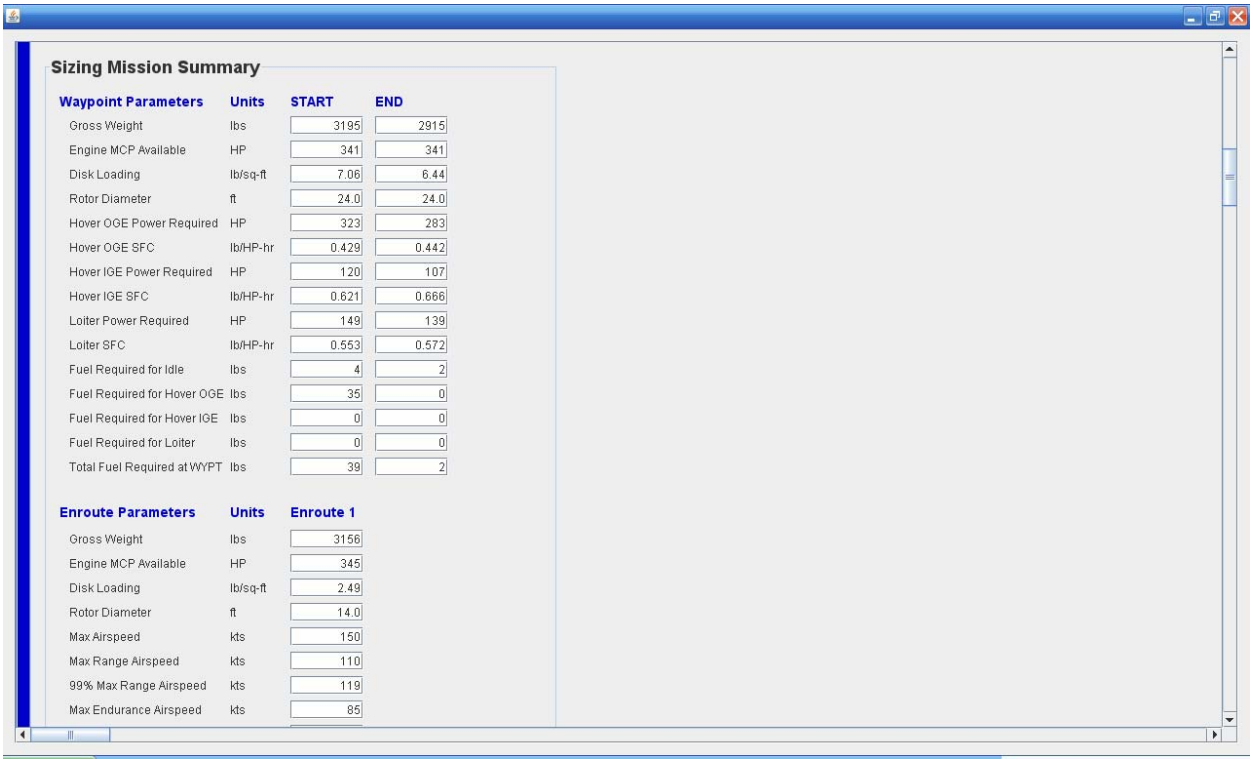


Figure 6.17: Sizing and Performance Summary (Bottom)

6.3 Calibration

The calculation methods of CIRADS were calibrated against the Hiller 1100 and the XV-15 with very accurate results. Figure 6.18 shows a max altitude vs. airspeed plot created with data from CIRADS for the Hiller 1100. This plot can be compared to the one of Figure 3.12 which is also for a Hiller 1100. Despite the jagged appearance (which is due purely to step size), the values are very close.

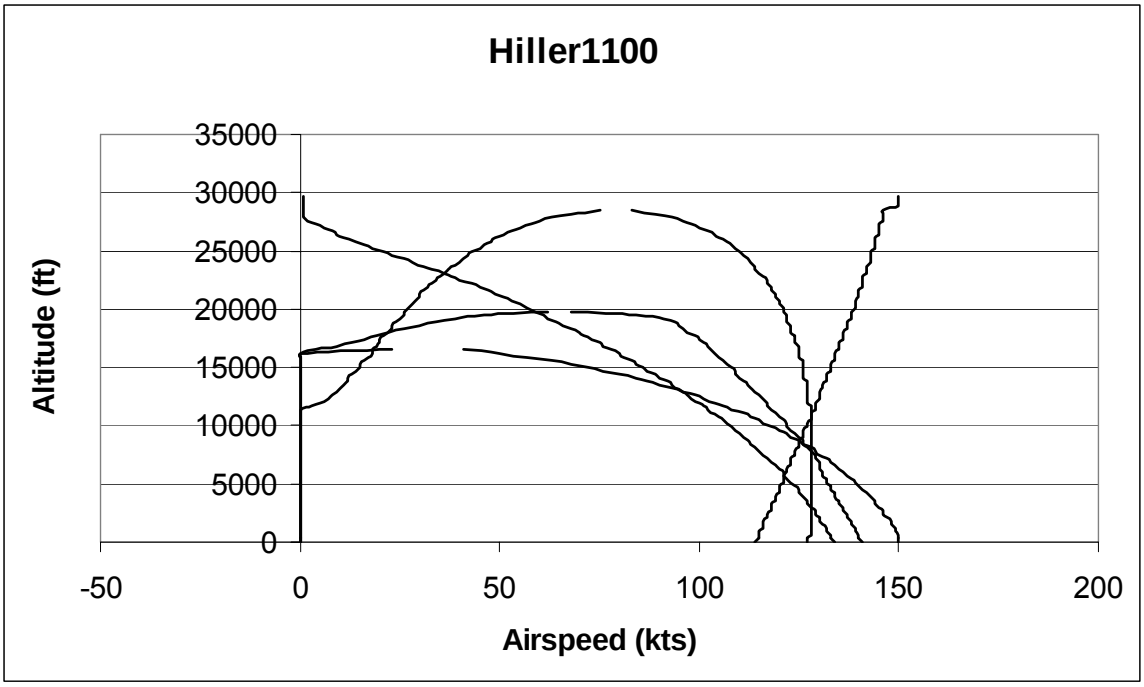


Figure 6.18: Hiller 1100 Max Altitude vs. Airspeed

A similar plot was created for the XV15 (Figure 6.19) and compared to the manufacturer's plot of Figure 6.20. Despite the obvious transition area between the helicopter and airplane mode (which CIRADS is not designed to model), the endpoints and critical values on the plot match considerably well.

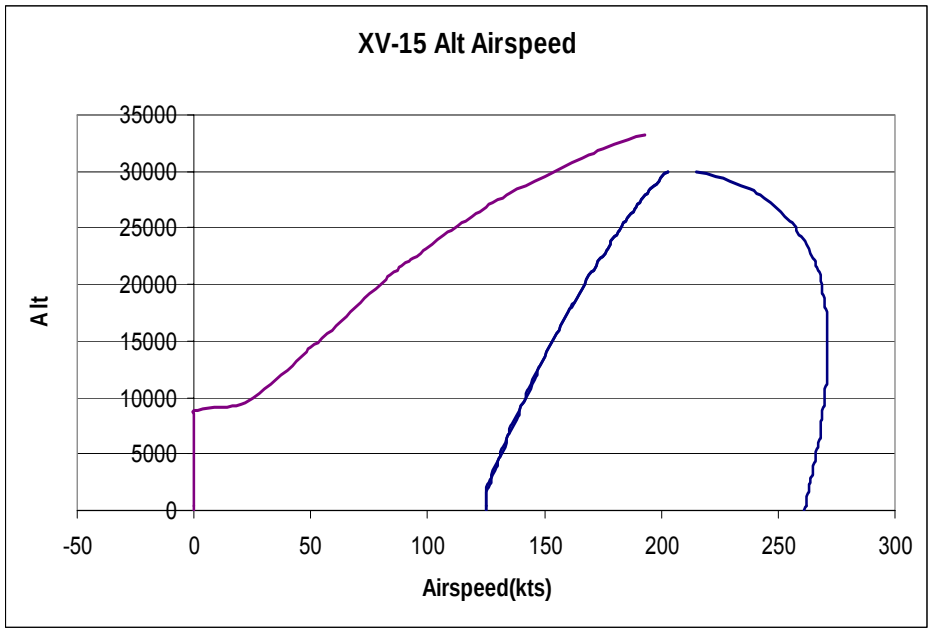


Figure 6.19: XV-15 Max Altitude vs. Airspeed (CIRADS)

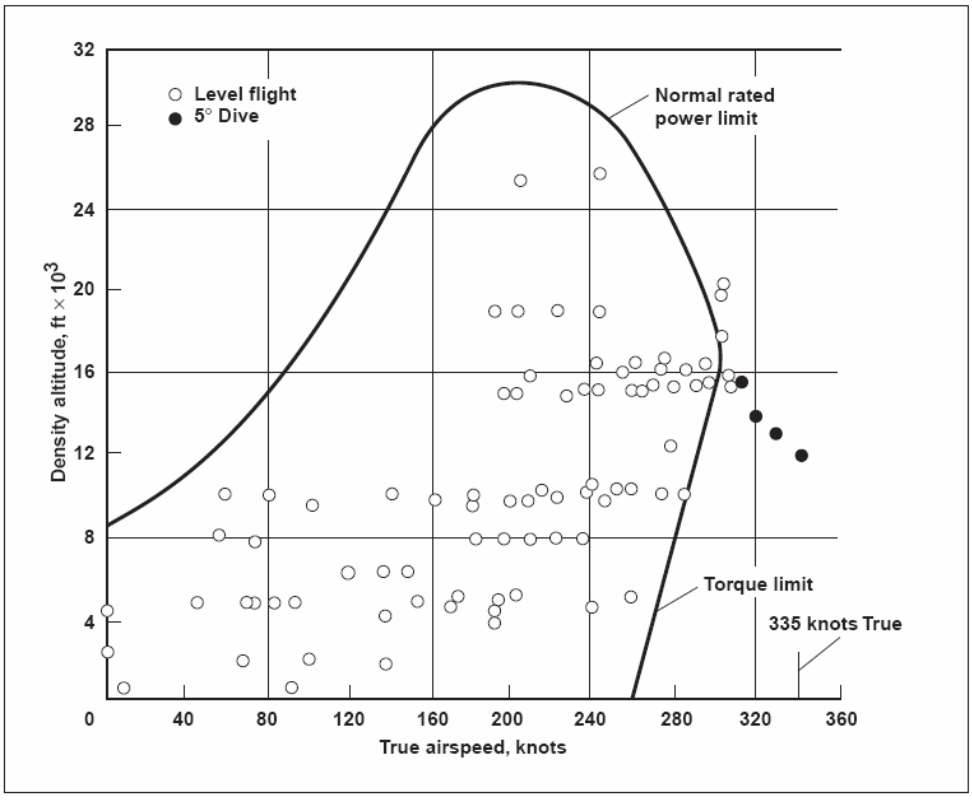


Figure 6.20: XV-15 Max Altitude vs. Airspeed (Actual)

CHAPTER 7: CONCLUSION

Based on the comparison of CIRADS results to the Hiller 1100 and XV-15 data, CIRADS is successful in modeling multiple rotorcraft configurations. The design cases for the 2007 AHS Student Design Competition (manned and unmanned) were also estimated with the Design Module and the gross weights were matched within 100 lbs for both cases which also demonstrates good correlation. There are still several features that can be added to CIRADS to make it more robust and user friendly, but this demo version gives a strong argument to the usefulness of modular conceptual design software.

REFERENCES

1. Prouty, R (1986) *Helicopter Performance, Stability, and Control*. PWS Publishers, Boston, Massachusetts

2. Maisel, M, Giulianetti, D, and Dugan, D (2000), The History of the XV-15 Tilt Rotor Research Aircraft: From Concept to Flight. National Aeronautics and Space Administration Office of Policy and Plans, Washington, DC

3. Schrage, D (1994) Extended R_F Methodology. Georgia Institute of Technology. Atlanta, Georgia

4. Leishman, G (2006) *Principles of Helicopter Aerodynamics*. Cambridge Press, New York, New York

5. Torenbeek, E (1982) *Synthesis of Subsonic Airplane Design*. Kluwer Academic Publishers, Dordrecht, The Netherlands

6. Sankar, L (2006) Rotary Wing Aerodynamics (AE6070) class notes. Georgia Institute of Technology, Atlanta, Ga.

7. Hiller Aircraft Corp (1960) Engineering Report Number 60-92: Proposal for the Light Observation Helicopter (Hiller 1100) Performance Data Report

8. 24th Annual AHS Student Design Competition 2007 Request for Proposal (RFP) for Advanced Deployable Compact Rotorcraft in support of Special Operation Forces Sponsored by Sikorsky Aircraft.